NEURAL NETWORK FEATURE RECOGNITION:

LOCATING THE CONIFER SEEDLING ROOT

COLLAR IN A DIGITAL IMAGE

By

MICHAEL PATRICK RIGNEY

Bachelor of Science
Oklahoma State University
Stillwater, Oklahoma
1985

Master of Science
Oklahoma State University
Stillwater, Oklahoma
1986

Submitted to the Faculty of the
Graduate College of the
Oklahoma State University
in partial fulfillment of
the requirements for
the Degree of
DOCTOR OF PHILOSOPHY
July, 1997

NEURAL NETWORK FEATURE RECOGNITION:

LOCATING THE CONIFER SEEDLING ROOT

COLLAR IN A DIGITAL IMAGE

Thesis Approved:

Glenn A. Kranzler

Thesis Advisor

Blayne E. Mayfield

Gerald H. Brusewitz

Thomas C. Collins

Dean of the Graduate College

ACKNOWLEDGMENTS

I wish to express my sincere appreciation to my major advisor, Dr. Glenn Kranzler, for his dedicated support and guidance. His patience and attention to detail were invaluable in the completion of my degree program. I wish to thank my advisory committee Dr. Gerald Brusewitz, Dr. Marvin Stone, and Dr. Blayne Mayfield for their generous time and efforts.

Thanks again to Dr. Glenn Kranzler and Dr. Gerald Brusewitz for your gracious efforts as my supervisors in my position of Research Engineer. I also thank the Department of Biosystems and Agricultural Engineering. I've had the privilege of working with a great group of people. Further, the security of a staff position has alleviated a great many of the anxieties which would normally accompany the pursuit of an advanced degree.

Grants supporting my work developing automated seedling inspection systems have been provided by the USDA Forest Service Missoula Technology & Development Center (MTDC) and the D.L. Phipps Nursery of the Oregon Department of Forestry. MTDC provided a grant specifically supporting this research.

Finally, I sincerely thank my wife, Michelle, for her constant support, patience, and motivational assistance. I thank my son too. Alex, the joy of watching you grow has provided wonderful relief from the efforts culminating in this dissertation. I dedicate this work to my wife and son, Michelle and Alexander, I love you always.

TABLE OF CONTENTS

Chapter Page

LIST OF FIGURES

Figure Page

LIST OF TABLES

CHAPTER I

INTRODUCTION

Well over one billion conifer seedlings are produced in the U.S. each year to support reforestation. Most seedlings are graded manually to improve viability after transplanting. Graders assess morphological characteristics which include stem diameter, shoot height, and root mass. Manual grading is labor intensive, subjective, prone to error, and increasingly costly. The large seasonal workforce has become difficult to recruit and retain in some regions of the U.S. These, and other concerns have sparked interest in automated seedling inspection systems. In addition to objective assessment of morphological features, automated inspection systems offer several advantages which are not feasible under current practice. These include multi-class grading and sorting, complete production statistics, and increased yield and value.

A PC-based line-scan machine vision system providing rapid measurement of bare-root seedling morphological features and multi-class grading has been developed by the author (Rigney and Kranzler, 1995). The system has demonstrated stem diameter measurement precision superior to that of nursery quality control personnel, and comparable precision for shoot height measurement, which has high variability. Precise morphological measurements require reliable identification of the seedling root collar, defined as the location at which the seedling stem intersects the soil surface. The large variability of seedling morphology makes automated root collar recognition a significant challenge. Root collar recognition is currently performed using a heuristic algorithm in the above automated inspection system. Location errors occur, and occasionally, the heuristic algorithm fails to find the seedling root collar.

Artificial intelligence techniques, and neural networks in particular, have yielded performance superior to traditional techniques in loosely constrained and noisy pattern recognition applications. Human pattern recognition often relies on qualitative and highly subjective criteria which are difficult to emulate with fundamentally quantitative computer-based systems. Neural networks are a class of computational algorithms modeled after the architecture and computing method of the brain. They are trained to provide the desired functionality through the presentation of input/output examples. The challenge of subjectivity and interacting factors inherent in image understanding and pattern recognition applications can be addressed more successfully with "intelligent" computational approaches such as artificial neural networks. Neural networks were therefore investigated as an alternative to the seedling inspection system's heuristic algorithm for recognition of the conifer seedling root collar.

The objectives of this research are presented in the next chapter. Manual and automated seedling inspection, the root collar recognition problem, and the heuristic algorithm are described in Chapter 3. Performance of human experts and the heuristic algorithm are analyzed, suggesting that an improved automated root collar recognition method is needed. An introduction to neural networks is provided in Chapter 4, followed by a discussion of feedforward multi-layer networks and the backpropagation training algorithm. Related neural network applications are reviewed. In Chapter 5, image processing and feature extraction techniques are described from the context of generating useful inputs to a neural network recognition system. This is followed by a discussion in Chapter 6 of the neural networks developed in this study. Network performance is discussed in Chapter 7. Conclusions are presented in Chapter 8.

Neural networks characterized by three different architectures were developed and tested. Among these networks, alternative input features, processing element configurations, and outputs were investigated. A total of 243 different networks and 57 input features were investigated. Individual networks used from 2 to 60 inputs and generated 1 or 4 outputs. Statistical analysis of network performance was conducted to identify the best configurations. Network performance is also compared with that of the heuristic algorithm.

CHAPTER II

OBJECTIVES

The primary objective of this research is to investigate neural networks for recognition of the root collar feature in digital images of conifer seedlings. The root collar is a critical reference point for seedling quality evaluation. Since network performance may be significantly affected by the input features extracted from digital images and by network architectural variations, a broad exploration of these variables is pursued.

It is desirable that neural network root collar recognition benefit the operational seedling inspection system. Therefore, a secondary objective is to explore network performance compared to that of the current heuristic algorithm technique. For superior performance, the neural network technique should provide a lower root collar location error rate than the heuristic algorithm.

Finally, consistent with support of real-time seedling inspection, a neural network recognition system with reasonably constrained input and architectural complexity is desired to limit the computational burden.

CHAPTER III

SEEDLING INSPECTION

Seedling Production and Quality Control for Reforestation

Well over one billion conifer seedlings are produced in the U.S. each year to support reforestation (Mangold et al., 1992). Most seedlings are graded manually to improve viability after transplanting. Grading is typically performed at grading tables and conveyor belts by inspectors who visually examine individual seedlings for dimension and appearance.

Morphological characteristics are used for grading nursery stock. These characteristics include stem diameter at the root collar (Fig. 1), shoot height and dry weight, root dry weight and volume, root fibrosity, foliage color, presence of terminal buds, shoot/root volume ratio, and ratio of top height to stem diameter (sturdiness ratio) (Forward, 1982; Ritchie, 1984; May, 1985). Stem diameter, shoot height, shoot/root ratio, and sturdiness ratio are generally given priority. Stem diameter is typically considered most important. Most of these characteristics are dependent on the location of the seedling root collar. Seedling dry weights and shoot and root volumes are time-consuming measurements; not generally assessed, except for research purposes. These characteristics are highly correlated with seedling projected areas measured by machine vision, thus allowing rapid assessment (Suh and Miles, 1988; Rigney and Kranzler, 1993).

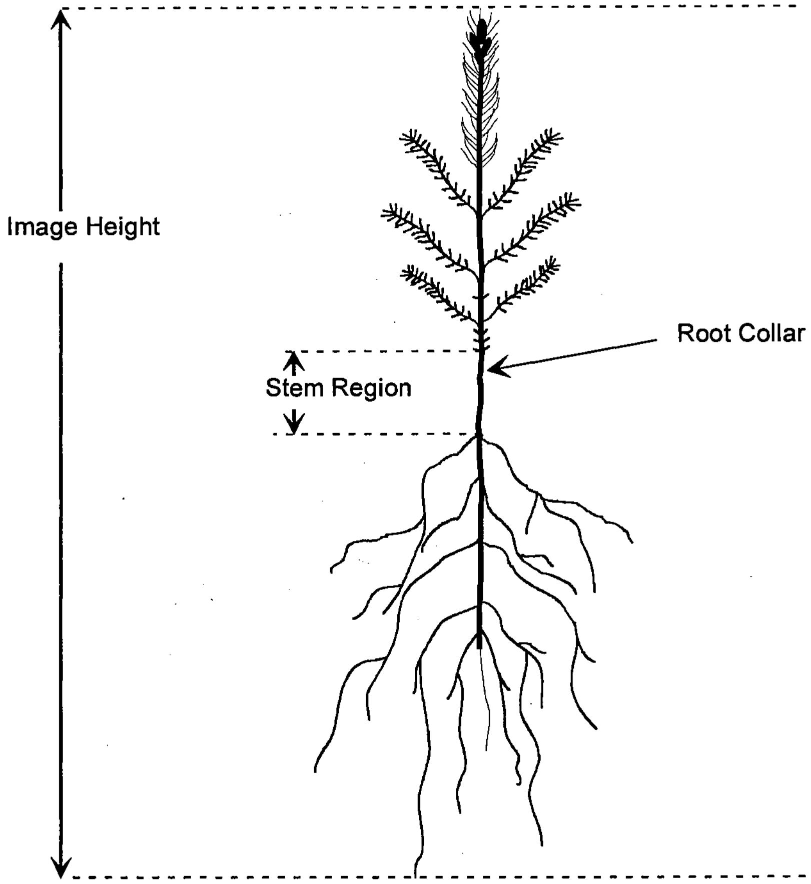


Figure 1. Seedling root collar, stem region, and image height.

Quality control assessments are used to ensure that graded seedlings meet specific grading criteria. Quality control at one USDA Forest Service nursery is performed by sampling at least 1% of seedlings from the grading tables and determining compliance with grading criteria (Scholtes, 1994). Culls are sampled to estimate loss of shippable product. The number of specific quality criterion failures per sample is the full extent of the quality record. Production demands do not allow measurement of sample morphology, which would better characterize quality.

Need for Automated Seedling Inspection

Interest in automated seedling inspection has been growing as a result of several factors. Manual grading is labor intensive and increasingly costly. Average annual production at 142 federal, state, and forest industry nurseries is 10 million seedlings (Okholm and Abriel, 1994). Seedling harvest is seasonal. Lifting and grading are typically performed within a two-to-three month (winter) window. Nursery production may be targeted at 100,000 to 250,000 seedlings per day.

Nursery production schedules necessitate a large temporary labor force. Hiring and retaining this labor force is becoming increasingly difficult. Minimum wage is often paid, however, higher wages are commonly needed to obtain enough workers. West coast and Canadian nurseries may pay graders in excess of $7.00 and $10.00 per hour, respectively. Grading is the largest component of seedling production costs. Lifting and grading account for two-thirds of the direct cost of seedling production at a typical nursery (Davis and Scholtes, 1995). Grading costs range from $1.30 to $6.70 per 1000 seedlings (Kranzler and Rigney, 1989). Other seedling production activities (bed preparation, sowing, fertilization, pest control, and lifting) have been automated.

Manual inspection requires sustained concentration. Classification is subjective and susceptible to human error. A low-temperature and high-humidity environment is used to reduce seedling stress, but is not conducive to human comfort. Repetitive, rapid hand motions make upper limb injuries a growing concern (Wallersteiner, 1988; Mowry, 1995). The grading task is complicated by the fact that grading criteria change for different species, age classes, seed lots, and customer preferences. Nurseries may grow over 650 different seed lots (Davis and Scholtes, 1995) and sell over 160 different stock types (Rose and Haase, 1995).

Benefits of Automated Inspection

In addition to supporting production needs at commercial nurseries, automated inspection offers several benefits which are impractical to obtain with manual grading. An automated system could sort seedlings into multiple classes defined for optimal performance at various planting sites, thus reducing mortality. For example, tall seedlings perform better on grassy sites and short seedlings perform better on droughty sites (Duddles and Owston, 1990). Cost and physical implementation difficulties prohibit manual multi-class sorting.

Customer specifications may easily be entered into the system and seedlings sorted accordingly. Increased yields may be realized by sorting and marketing alternate grades, using seedlings normally culled under current practice. These capabilities can increase both nursery yields and seedling value.

Customers can be provided with a statistical description of the seedlings purchased, and packages labeled accordingly. This information could be used for final decisions in assigning specific seedlings to planting sites (Scholtes, 1994). Accurate seedling package counts are an important benefit. Package count could easily be set to customer specification.

Comprehensive production statistics are often cited as the most valuable benefit of machine vision inspection systems. Applied to nursery and forest management, morphology statistics can be correlated with seed source, cultural practices, weather, and ultimately, field performance.

Automated Seedling Inspection

The feasibility of using machine vision for conifer seedling quality inspection has been demonstrated through a series of prototype systems. Rigney and Kranzler (1988, 1989) developed the first system, which used two video cameras, strobe lighting, and a dedicated high-performance machine vision computer to automatically locate the seedling root collar and measure stem diameter, shoot height, shoot area, and root area at a rate of two seedlings per second. The use of matrix CCD cameras and front-lighting limited spatial resolution and measurement precision.

Performance of a second prototype was enhanced through the use of backlighting and a single line-scan camera (Rigney and Kranzler, 1993). Using a high-performance VME-bus based computer and image processing hardware, this prototype achieved increased measurement precision and inspection rates of up to 10 seedlings per second. System cost, however, was high.

Wilhoit et al. (1994) describe two-camera and three-camera machine vision systems developed for seedling quality control and research measurement applications.

The two- and three-camera systems provided inspection rates of 5.8 and 15.8 s per seedling, respectively. Both systems had 0.1-mm resolution for diameter measurement. Root collar location was determined by the operator, and the root collar was positioned over a fixed reference when placing each seedling in front of the cameras. Seedling top location was indicated by a movable reference. These systems made no attempt to automatically identify the root collar or seedling top. Test results showed accurate diameter measurement. Low correlation between machine and manual height measurements was observed, however, low correlation was also observed among repeated manual measurements.

In recent work,. a PC-based line-scan machine vision system was developed for nursery quality control and morphological data acquisition (Rigney and Kranzler, 1995). A prototype is currently in use at a commercial nursery. The system shows strong promise for automating production-line grading. The machine provides a user-friendly, menu-driven graphical interface. Individual seedlings are manually placed on a conveyor belt and inspected in less than 0.25 seconds. The machine automatically locates the root collar and measures stem diameter, shoot height, sturdiness ratio, root mass length, projected shoot and root area, shoot-root area ratio, and percent fine roots. Sample statistics are computed for each measured feature. Measurements for each seedling may be stored for later analysis. Feature measurements may be compared with multi-class quality criteria to determine sample quality or to perform multi-class sorting. Statistical summary and classification reports may be printed to facilitate the communication of quality concerns with grading personnel.

Several related aspects of defect detection and seedling handling have been investigated, targeting the development of a comprehensive automated system. Research projects have addressed color defect detection, seedling sorting, and seedling accumulation/handling for presentation to a root pruner. A high-speed, reciprocating-brush seedling sorter was successfully developed and demonstrated (Kranzler and Rigney, 1994). Post-sorting seedling handling was investigated and prototype systems were constructed and tested. A pinch-twist conveyor, which accepts horizontally positioned seedlings from the sorter and rotates them to a vertical orientation, was developed to support on-line root pruning. Further software and equipment development efforts addressed color defect detection (foliage chlorosis, stripped root laterals), and root pruner integration (Kranzler and Rigney, 1996). After pruning, a software controlled accumulation conveyor groups seedlings for packaging.

Locating the Seedling Root Collar

Precise morphological measurements and accurate grade assignment require reliable identification of the seedling root collar location (Fig. 1). The root collar is a critical reference for the measurement of several seedling features (diameter, shoot height, root length, shoot/root ratio, and sturdiness ratio). Root collar recognition is the most challenging aspect of machine vision seedling inspection. This function is currently achieved using a heuristic algorithm which relies on many operator-controlled parameters to extract root collar location cues based on seedling shape. Although it is functional, the current root collar location technique is not optimal. Root collar location errors occur, and occasionally the heuristic algorithm fails to detect the seedling root collar.

Many definitions for the root collar location can be found in the literature (Menes and Mohammed, 1995). The root collar is commonly defined as the point (region) along the tree seedling stem at which the seedling intersects the soil surface. This location is difficult to determine after lifting (removal from the soil). In practice, nurseries specify an alternate position (which varies among nurseries) to be used as the root collar. The root collar is often defined to be located at the cotyledon scar or one cm above the uppermost root lateral (Mexal and Landis, 1990). From the author's observation at several nurseries, the root collar has been defined as a position up to 10 mm above or below the cotyledon scar.

Visibility of the cotyledon scar varies greatly among seedlings, from practically non-existent to an obvious girdled bulge on the stem. The cotyledon scar is located at or below the lowest needles and/or branches, and above the uppermost root laterals. The length of the "stem region" void of needles, branches, and root laterals, can vary from 1 to 10 cm or more. Due to morphological variation, the cotyledon scar may be located anywhere within the stem region, but is usually near the top. Stem diameter often tapers slightly, below the cotyledon scar.

The large variability of seedling morphology makes automatic root collar recognition a significant challenge. Within a sample, stem diameters can range from 2 to 13 mm, shoot heights from 10 to 40 cm, and root mass lengths from 15 to 40 cm.

Attached soil aggregates, complex branching, and root entanglement may obscure the root collar.

Current Approach - Heuristic Algorithm

The current seedling inspection system uses a heuristic algorithm to locate the root collar with a success rate of 70 to 90%, depending on seedling morphology. Seedling images are observed to have low complexity at the root collar and relatively greater complexity in the foliage and root zones. In the ideal case, image lines near the root collar contain only the seedling stem. Image lines above and below the root collar contain many needles and branches or many roots.

On the basis of these observations, a heuristic function which should have a local minimum near the root collar is defined. Features are extracted from each line in the image, multiplied by weighting factors, and summed, generating a value for each line in the image. The line features used in the heuristic function include: number of runs, area, and span. These features are described in the section on seedling image processing. After smoothing, this set of values may be interpreted as a function (signal) with many peaks and valleys. Through appropriate selection of weighting factors, the root collar will be located at a local minimum (valley) of the curve. Further processing is used to search for and select the root collar valley. Once selected, the seedling stem is located (horizontally), and the stem region is delineated (vertical limits). The root collar is taken as a location defined as a percentage along the stem region.

The large number of operator-defined parameters used for root collar location and the complexity of the heuristic algorithm give the system flexibility, but also make it difficult to configure. The weighting factors applied to the various features are difficult to optimize, and other useful features (with associated weights) could be defined, making weight optimization more challenging. Further, different conifer species and age classes have unique morphology which should be exploited through different optimal weights.

As previously mentioned, the heuristic algorithm occasionally fails to detect the seedling root collar. Usually, a segment of bare stem above or below the root collar is located. Needles, a small branch or root lateral, or debris may lie between the selected stem segment and the root collar. The stem region is generally visible in the seedling image under these circumstances. Needles, branches, or roots often parallel the stem region when there is a collar location failure. However, the heuristic algorithm often correctly locates the stem region in similar situations.

The performance of the heuristic algorithm described above is noteworthy, and the system is considered successful and useful for quality control and interactive morphological research. When used interactively, an operator may discard an erroneous measurement and re-scan the seedling. Improved root collar recognition performance is necessary, however, for unsupervised, 100% production-line inspection.

Previously Measured Performance

Results of previous studies illustrate the difficulty of precisely locating the seedling root collar and the need for an improved method. Although neither investigation directly compared machine and manual root collar identification, collar location differences and variance are closely related to the measured parameters; stem diameter and shoot height. Stem diameter is measured at the root collar. Seedling height is defined as the distance from the root collar to the end of the terminal bud. Some inspection system height measurement error may be associated with difficulty in locating the terminal bud.

A seedling inspection system (Rigney and Kranzler, 1993) was tested to compare manual measurements with those obtained by machine vision. Manual measurements were obtained by experienced nursery personnel. Stem diameter measurements were acquired directly above the cotyledon scar with a digital caliper and recorded to the nearest 0.1 mm. Shoot height was measured manually from the top of the terminal bud or leading shoot tip to the cotyledon scar with a meter stick and recorded to the nearest millimeter.

Machine vision measurement of each seedling was repeated either five or ten times. Seedlings were rotated about their longitudinal axis between repetitions so that a variety of projected views was acquired. Although this procedure increases variance, it provides a more realistic estimate of measurement precision.

Diameter and height of 80 two-year old Douglas fir seedlings were measured by three individuals. Ten repeated measurements were acquired by the vision system. Diameter and height of an additional 80 seedlings were each measured by four individuals. Five repeated measurements were acquired by the vision system.

Summary statistics are presented in Table 1. Means are those of the sample populations. Standard deviations are for repeated measurements, not the sample populations. The machine vision system occasionally failed to find the root collar or identified an incorrect location. Those observations were omitted from the analysis (anticipating improvements!)

Table 1. Summary statistics for seedling measurements.

		Machine[a]			Manual[b]			Difference[c]	
Feature	Unit	Mean	Stdev	CV%	Mean	Stdev	CV%	Mean	Stdev
Diameter	mm	5.1	0.1	1.9	5.0	0.3	6.5	0.1	0.5
Height	mm	349	7	2.1	338	18	5.3	13	21

[a] - 1140 observations from 160 seedlings

[b] - 560 observations from 160 seedlings

[c] - machine mean - manual mean, 160 observations

Machine diameter measurement standard deviation (0.1 mm, Table 1) is one-third that of manual measurement and compares favorably with the lateral spatial resolution of 0.25 mm. Machine height measurement standard deviation was 7 mm, much greater than the longitudinal resolution of 1 mm per line, but compares favorably with manual measurement standard deviation of 18 mm.

Machine and manual measurements were compared for each seedling. Machine diameter measurements were larger than manual measurements by an average of 0.1 mm, with a standard deviation of 0.5 mm. The standard deviation of the measurement differences is 10% of the average seedling diameter. Machine measured height was larger than manually measured height by an average of 13 mm, with a standard deviation of 21 mm. This standard deviation is 6% of the average seedling height.

A later test was conducted using the PC-based inspection system (Rigney and Kranzler, 1995) to compare machine and manual seedling measurements. One hundred seedlings each of two-year old Ponderosa pine and Douglas fir were measured four times each by the vision system and once each by four different quality control personnel. Manual diameter measurements were obtained with a digital caliper and recorded to the nearest 0.1 mm. Manual height and root mass length measurements were obtained with a meter stick and recorded to the nearest millimeter. Seedlings were rotated about their longitudinal axis between consecutive machine vision measurements.

Table 2 provides a comparison between machine and manual measurement precision. Machine diameter measurement standard deviation was 0.09 mm, less than one-fourth that of manual measurements. Machine and manual measurements had similar variation for height.

Table 2. Machine vs. manual measurement performance.

		Stdev		Machine - Manual	
Feature	Unit	Machine	Manual	Mean	Stdev
Diameter	mm	0.09	0.41	0.34	0.53
Height	mm	8	7	37	22
Rt. Mass Len.	mm	14	11	-19	20

200 seedlings, 4 reps.

The difference between manual and machine measurements is also presented in Table 2. Machine diameter measurements were, on average, 0.34 mm larger than manual measurements. The standard deviation of measurement differences indicates that the two methods could easily differ by one millimeter. Most of this "error," however, may be attributed to the manual measurements, which had much lower precision. Machine height measurements were significantly larger than manual measurements. This result is partially due to difficulty in locating the terminal bud on many seedlings. The vision system also tended to locate the root collar lower on the stem than did the quality control personnel. The difference between manual and machine root collar location was partially responsible for shorter machine measurements of root mass length.

These tests show superior diameter measurement precision for the machine vision system. Mean diameter measurement differences may result from the different root collar locations selected by the machine and manual graders. Disagreement on root collar location is more apparent in the height measurement statistics. Improved root collar location should increase the accuracy of all machine vision measurements.

Root collar location was manually identified in seedling images used in this study. Thus, a direct evaluation of heuristic algorithm performance was obtained. Heuristic algorithm performance results are presented later with neural network results. Measured performance was consistent with height measurement precision discussed above.

CHAPTER IV

NEURAL NETWORKS

Neural networks are most frequently applied to classification, pattern recognition, and prediction. These application categories are broad, however, and overlap. Pattern recognition often relies on qualitative and highly subjective criteria which are difficult to emulate with fundamentally quantitative computer-based systems. The large variation and interacting factors inherent in pattern recognition of agricultural products may be addressed more successfully with "intelligent" computational approaches such as artificial neural networks.

In this chapter, a brief overview of neural networks is initially presented. Next, the computation used in multi-layer, feedforward networks and the backpropagation training algorithm, which were used in this study, are reviewed. Finally, several image pattern recognition applications are surveyed and their relationship to the problem of root collar recognition is discussed.

Introduction to Neural Networks

Artificial neural networks are a class of computational algorithms based on the architecture and computing method of the brain (Anderson, 1988). Unlike conventional algorithms and other artificial intelligence techniques, neural networks do not store data. They encode knowledge through the connectivity and weights between processing elements (PE). A neural network is trained (weights adjusted) to produce the desired behavior through presentation of examples. Network specifics are application-dependent. Neural networks vary in number of PE's, connection architecture, computation algorithm, and training algorithm.

Neural models have the following general characteristics (Rumelhart et al., 1986a): 1) a set of processing units, 2) a state of activation of the units, 3) a rule for combining unit inputs, 4) a transfer function computing an output for each unit, 5) a pattern of connectivity among the units, 6) a propagation rule for relaying unit outputs through the network, 7) a learning rule for modifying unit connectivity and/or connecting weights, and 8) an environment in which the system operates (input/output). Many different types of neural networks have been defined within this model. These types include the perceptron and multilayer perceptron, Hopfield, Kohonen, adaptive resonance theory, and the dynamic link architecture, all of which have been used for pattern recognition (Wang, 1993) among other applications.

The multilayer perceptron is one of the most frequently used and well studied neural network models (Carpenter, 1989). It is a multilayer feedforward network, also called the backpropagation network, due to its training algorithm (Rumelhart et al., 1986b). These networks consist of an input layer, one or more hidden layers, and an output layer. Each layer is composed of simple PE's which model neurons. A PE has one or more inputs and a single output. PE output is computed by multiplying each input with an associated weight, summing the products, and adding a bias value. The result is scaled between 0 and 1 (or -1 and 1) by a transfer function to produce the PE output. The output may be used as input to any number of PE's in the next layer. In operation, a pattern is presented to the input layer of the network. The process described above is used to feedforward a pattern of activation from layer to layer until the output layer is reached. The activation (values) of the output PE’s is the result of the network computation.

Many advances have been made in basic and applied neural network research over the last ten years. Much emphasis has been placed on the development and enhancement of training algorithms to improve performance and reduce training times (Weymaere and Martens, 1991; Romaniuk and Hall, 1993; NeuralWare, 1995a). Hardware implementations have also been pursued to obtain faster network evaluation and training. The effect of training sample size on recognition performance has received some attention (Baum and Haussler, 1989; Raudys and Jain, 1991). Much application-specific research has investigated the effects of varying network architecture and feature representation on network performance.

The Backpropagation Algorithm

The backpropagation algorithm is presented here, beginning with the feedforward computation performed to produce outputs from network inputs (Fig. 2). Each PE is generally connected to every PE in the next layer. Layers are denoted by the superscript, s, and numbered starting at the input layer (s = 0), through hidden layers to the output layer (s = a).

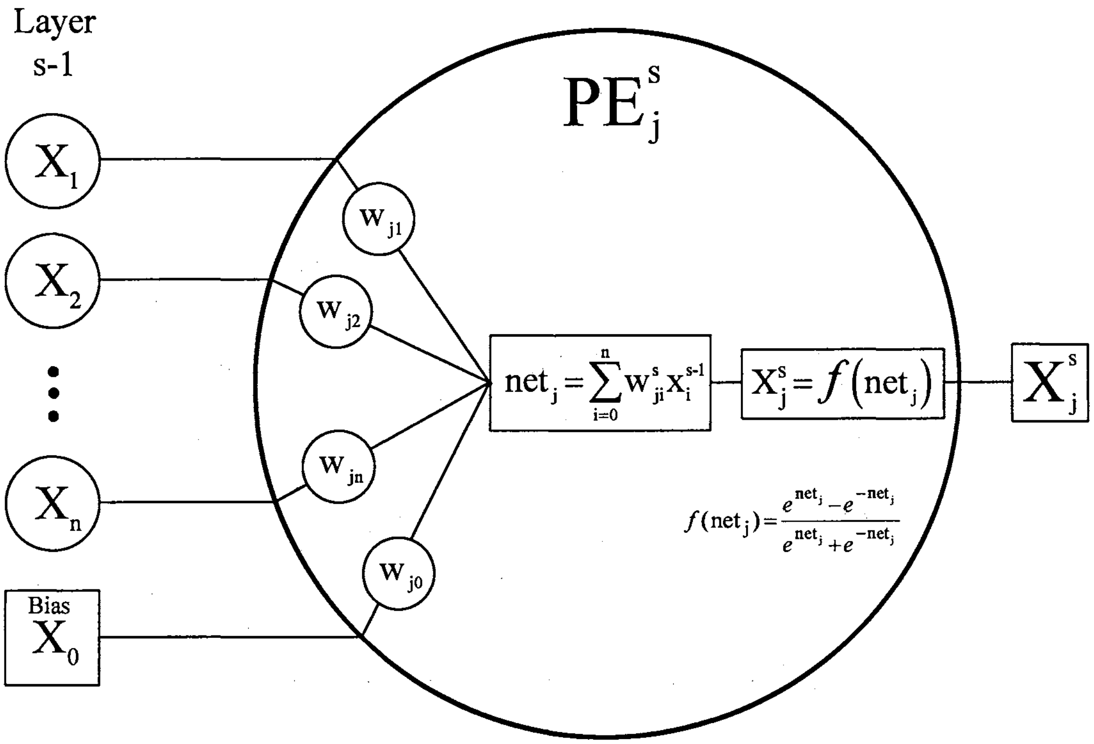

Layer
s-1
PE_j^s
X_1
X_2
X_n
Bias
X_0
w_{j1}
w_{j2}
w_{jn}
w_{j0}
$net_j = \sum_{i=0}^{n} w_{ji}^s x_i^{s-1}$
$X_j^s = f(net_j)$
X_j^s
$f(net_j) = \frac{e^{net_j} - e^{-net_j}}{e^{net_j} + e^{-net_j}}$

Figure 2. Feed-forward computation of processing element activation for PEj in neural network hidden or output layer s.

The net input to a PE is expressed in Equation 1:

$$net_j^s = \sum_{i=0}^{n} w_{ji}^s x_i^{s-1} \tag{1}$$

where

net_j^s = net input to the *jth* PE in layer s, for $0 < s \leq a$
x_i^{s-1} = output of the *ith* PE in layer s-1
w_{ji}^s = weight multiplier between the *jth* and *ith* PE's

A single bias PE is connected to all PE's in the hidden and output layers. The bias input is denoted as the 0*th* PE (i = 0) in Equation 1, and has a constant value of 1. PE bias is adjusted through the bias connecting weight, w_{j0}. The remaining n inputs to a PE are the outputs of PE's in the previous layer.

Output of a PE is computed using a transfer function as in Equation 2:

$$x_j^s = f(\text{net}_j^s) \tag{2}$$

A commonly used transfer function is the sigmoid, which produces an output asymptotically approaching 0 and 1 for large negative and positive net inputs, respectively:

$$f(\text{net}_j) = (1.0 + e^{-\text{net}_j})^{-1} \tag{3}$$

Use of the hyperbolic tangent transfer function is also common. It was selected for networks developed in this study. The hyperbolic tangent produces an output approaching -1 and 1 for large negative and positive net inputs, respectively:

$$x_j^s = f(\text{net}_j) = \tanh(\text{net}_j) = \frac{e^{\text{net}_j} - e^{-\text{net}_j}}{e^{\text{net}_j} + e^{-\text{net}_j}} \tag{4}$$

Thus, equations 1 and 4 are used to compute PE outputs, starting at the first hidden layer and progressing toward the output layer. To train a network, we adjust the weights, w_{ji}, to reduce the error between the network output (vector), $\mathbf{x}^a$, for a given input, **i**, and the desired or target output, **t**, for that input. The error at the network output is computed as:

$$E = \frac{1}{2}\sum_k (t_k - x_k^a)^2 \tag{5}$$

where *k* denotes the components of the output and target vectors.

Weights are adjusted using the generalized delta rule, which performs a gradient descent in the weight space. The change to a given weight is proportional to the change in the error with respect to the change in the weight (gradient):

$$\Delta w_{ji}^{s} = -\eta \frac{\partial E}{\partial w_{ji}^{s}} \tag{6}$$

where the constant of proportionality, η, is called the learning rate, and the negative sign provides gradient descent. Learning rate could be written, η^{s}, since it can vary for different layers. Derivation of the error gradient, $\frac{\partial E}{\partial w_{ji}^{s}}$, is described in detail by Rumelhart (1986b) and Pao (1989). The result is that the weight change is proportional to the product of an error component, δ^{s}, at PE_{j}^{s}, which receives input through the weight, and the output of the $PE_{j}^{s\text{-}1}$ sending activation through the weight (Fig. 3). Thus, Equation 6 is rewritten:

$$\Delta w_{ji}^{s} = \eta \delta_{j}^{s} x_{i}^{s\text{-}1} \tag{7}$$

The error component, δ^{s}, is computed differently for PE's in the output and hidden layers. Output layer PE error is readily computed, given the target output vector, **t**. For each output PE_j:

$$\delta_{j}^{a} = (t_{j} - x_{j}^{a}) f'(net_{j}^{a}) \tag{8}$$

where $f'(net_{j}^{s})$ is the derivative of the transfer function evaluated at the value of the net input to the PE. The derivative of the hyperbolic tangent transfer function is:

$$f'(net_{j}) = \mathrm{sech}^{2}(net_{j}) = \left(\frac{2}{e^{net_{j}} + e^{-net_{j}}}\right)^{2} \tag{9}$$

which can also be expressed as a function of the transfer function itself:

$$f'(\mathrm{net_j}) = (1 + f(\mathrm{net_j})) \cdot (1 - f(\mathrm{net_j})) \tag{10}$$

or as equivalent, substituting Equation 2:

$$f'(\mathrm{net^s_j}) = (1 + \mathrm{x^s_j}) \cdot (1 - \mathrm{x^s_j}) \tag{11}$$

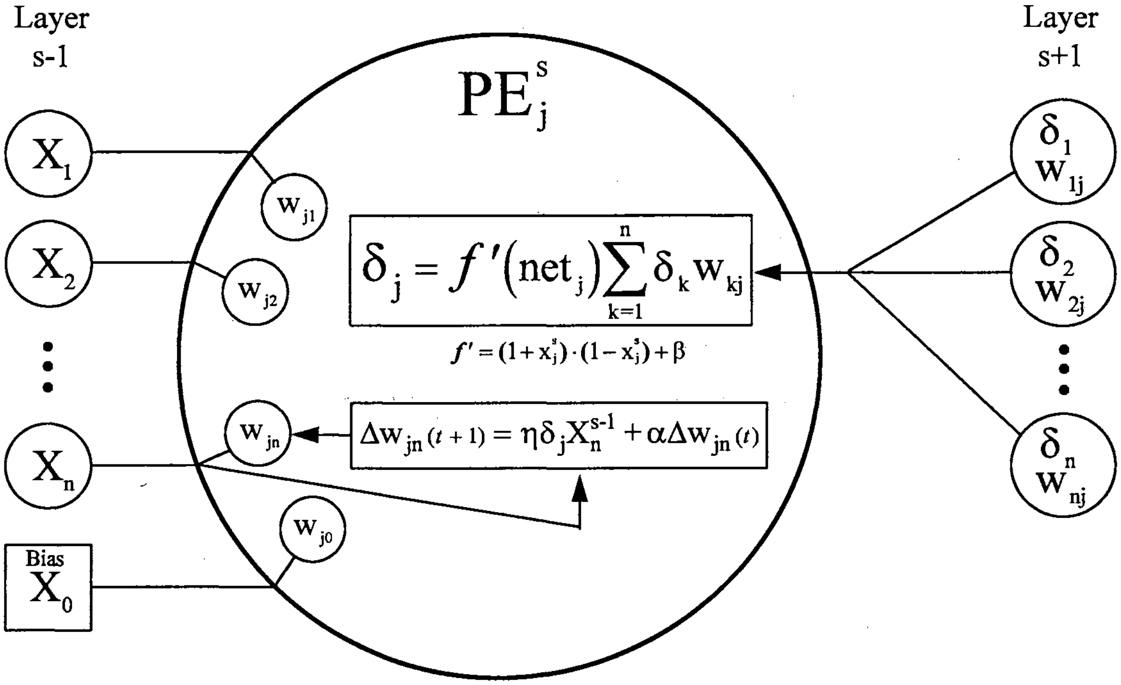

Layer s-1
Layer s+1
PE^s_j
X_1
X_2
X_n
Bias
X_0
w_{j1}
w_{j2}
w_{jn}
w_{j0}
$\delta_j = f'(\mathrm{net_j})\sum_{k=1}^{n}\delta_k w_{kj}$
$f' = (1+x^s_j)\cdot(1-x^s_j)+\beta$
$\Delta w_{jn}(t+1) = \eta\delta_j X^{s-1}_n + \alpha\Delta w_{jn}(t)$
δ_1 w_{1j}
δ_2 w_{2j}
δ_n w_{nj}

Figure 3. Backpropagation algorithm for computation of weight adjustment in PEj of neural network hidden layer s.

Error components for hidden PE's are computed from the error components of the PE's in the next layer; they are back-propagated from the output layer toward the input layer:

$$\delta^s_j = f'(\mathrm{net^s_j})\sum_k \delta^{s+1}_k \mathrm{w}^{s+1}_{kj} \tag{12}$$

Net PE activation can become large when weights become large during training (Eq. 1). PE output (Eq. 2) will then become saturated (-1 or +1). Thus, the transfer function derivative approaches zero (Eq. 11), as do the error components (Eq. 8 and 12), so no learning takes place. A small positive offset, β, is often added to the transfer function derivative to increase learning speed (NeuralWare, 1995a). Equation 11 may be rewritten:

$$f'(\mathrm{net}_j) = (1 + \mathrm{x}_j^s) \cdot (1 - \mathrm{x}_j^s) + \beta \tag{13}$$

A small learning rate is needed to achieve true gradient descent. This rate, however, results in slow learning, and weights may converge to a local minimum of the error function (surface). Large learning rates may produce oscillation in the weight space. A momentum term is commonly added to Equation 7 to overcome these difficulties:

$$\Delta \mathrm{w}_{ji}^s(n+1) = \eta \delta_j^s \mathrm{x}_i^{s\text{-}1} + \alpha \Delta \mathrm{w}_{ji}^s(n) \tag{14}$$

where α is referred to as the momentum coefficient, or simply momentum. Thus, the weight change at a given iteration is proportional to the weight change at the previous iteration.

Error components and associated weight changes discussed above are computed on the basis of individual input-output training examples (Eq. 5). Weight changes based on the global error, among all training examples, can be approximated by accumulating weight changes for a number of training examples (epoch), before implementing the weight change:

$$\Delta \mathrm{w}_{ji}^s(n+1) = \sum_{k=1}^{\mathrm{epoch}} \left(\eta \delta_j^s \mathrm{x}_i^{s\text{-}1} \right) + \alpha \Delta \mathrm{w}_{ji}^s(n) \tag{15}$$

When this approach is taken, learning rate should be adjusted according to epoch size in order to control the magnitude of the accumulated weight change.

To summarize, the backpropagation algorithm is implemented by recursively computing feedforward activation (Eq. 1 and 4), then backpropagating error components and computing weight updates. Backpropagation is implemented by four equations. The transfer function derivative is computed at each PE using Equation 13. Error components for output PE's are computed using Equation 8. Error components for hidden layer PE's are computed layer by layer, from output toward input, using Equation 12. Weight changes are accumulated for epoch training examples, and then updated using Equation 15.

Image Pattern Recognition Applications

Neural networks are commonly used for classification, pattern recognition, and prediction. Network inputs represent measured features of an object obtained through conventional sensing techniques. These measurements are composed into a feature vector. For classification, the network assigns the feature vector (object) to one of a number of classes.

Pattern recognition differs from classification, in that input data may be more abstract and/or partially missing. Recognition of printed or handwritten characters, and faces are two examples. In prediction applications, network input is a series of previous values of a variable(s), and the output is the current or future value. Financial market forecasting and process modeling/control are two examples.

Digital image processing and analysis operations generate large quantities of data. A single pine seedling image, for example, can occupy two megabytes of memory. Image enhancement operations (filtering, edge detection) may generate new images equal to the original in size. Many operations are performed to extract features from an image. Such data reduction operations are crucial to conventional image analysis and interpretation algorithms. Extracted features, however, are often difficult to interpret, and object recognition or classification may depend on interacting relationships among features. Traditional statistical analysis and classification techniques are usually applied in these cases, often with less than optimal results.

Ghazanfari and Irudayaraj (1994) used four independent neural networks to recognize four classes of pistachio nuts (3 varieties, 1 un-split class). Five features were extracted from silhouette images (area, roundness, perimeter, minimum, and maximum radius). Each neural network (NN) had 5 inputs, 5 and 3 PE's in the first and second hidden layers, respectively, and a single output. Nut classification was associated with the network producing the maximum output. Networks were trained using 100 nuts and tested on an additional 100 nuts from each class. Overall classification accuracy was 89%.

Park et al. (1994) used a backpropagation network to process spectral features extracted from beef steak ultrasound images. The Fourier transform was applied, and seven spectral features were extracted: lower, central, and upper frequency, peak frequency, bandwidth, skewness, and number of local maxima. These features were used as inputs to networks which predicted the human sensory attributes: juiciness, fiber and overall tenderness, flavor intensity, and connective tissue amount. NN prediction error (rms) averaged 0.12 for all sensory attributes which were rated on a scale of 1-8. Application of network outputs to two-class quality grading yielded 83% to 75% accuracy for the various quality attributes.

Burks et al. (1994) used a backpropagation network to recognize seven plant species from canopy images. RGB color images were transformed to the HSI color space. Spatial Gray-level Dependence Matrices (SGDM, co-occurrence matrix) were computed from the HSI bands to quantify image texture. Eleven texture features were then extracted from the SDGM of each band. These included mean, variance, correlation, moments (3), and entropy (3). Subsets of the 33 features were used as inputs to various neural networks. Recognition accuracy ranged from 76% for all species, to 96% for four dissimilar species.

Vincent (1995) described a Hierarchical Perceptron Feature Locator (HPFL) applied to finding faces in digital images. Analysis was constrained to head-and-shoulders images with limited variation of facial orientation. The hybrid system was composed of a large number of Multi-Layer Perceptron (MLP) feature detectors. Initial 256x256-pixel gray level images were compressed to 16x16 pixels. MLP feature detectors (5x5 inputs; right eye, left eye, mouth, and face center) were scanned over the image. Results accumulated in feature maps were post-processed geometrically and thresholded to reduce false-positive feature locations. Tiles from the initial image (16x16 pixels) corresponding to potential feature locations were assembled into a frame.

High-resolution detectors composed of several MLP's (11x9 pixel inputs) were then scanned across the frame. Four MLP's for the mouth, for example, detected the right and left edges of the mouth, the top of the upper lip, and the bottom of the lower lip. These detector outputs were post processed to yield the feature location. The system was trained on sixteen images. On a test set of 44 images, facial features were located with an average error of 0.73 pixels and a maximum error of 2 pixels.

Golomb and Sejnowski (1995) describe a system to recognize (classify) the gender of a person given a facial image. Ninety gray level images (512x512 pixels) of young adult faces (45 male, 45 female) with no facial hair, jewelry, or makeup were used in the study. Preprocessing normalized eye and mouth locations. Background, clothing and most hair were masked. Images were then compressed to 30x30 pixels. A hetero-associative network with 900 inputs (30x30 pixels) and 40 hidden layer PE's, was used to obtain a further compressed (hidden layer) representation.

The compressed representation was input to two gender classification networks, having 0 and 40 hidden layer PE's, respectively. A single binary output classified faces as male or female. Networks were trained using backpropagation on sets of 80 images and tested on the remaining 10. The networks had similar performance, correctly classifying 89% of the images. Five humans, viewing the original 512x512 pixel images, correctly classified 88% of the images. A 900 input network (30x30 image) also provided good performance. Compression to 40 inputs, however, greatly reduced training time.

Casasent and Sipe, (1997) used a piecewise quadratic neural network to process features extracted from x-ray images of pistachio nuts for detection of worm damage. The network itself was a modification of the multilayer perceptron which used complex-valued weights between the input and hidden layer PE's, and a magnitude-squared nonlinearity in the hidden layer PE's (Casasent and Natarajan, 1995). A special algorithm was used for error backpropagation training. Initial image processing was performed to segment the nutmeat image from edge and shell image components. Histograms of the gray-scale and edge-enhanced nutmeat images were then computed. The histogram is naturally rotation invariant. Histogram normalization was used to obtain scale invariant features. Histograms were compressed (8 to 20 bins) and statistical features (mean, variance, skew, kurtosis) of the histogram were computed. Histogram data and statistical features were used as network inputs. Networks with various input features and hidden layer PE's were investigated. The best network yielded 89% correct classification, corresponding to 14% rejection of good product . A confidence level computation applied to the network outputs allowed performance to be adjusted such that 98.7% of the good product was retained, while the final mix contained only 1.3% damaged nuts.

Interesting and dissertation project-related properties of these applications are summarized here. The original images processed contain large amounts of information. Image compression, transformation, and feature extraction were used to reduce amount of data processed by the neural networks. Some of these techniques were also used to obtain translation and rotation invariant recognition. Data compression reduced NN training and evaluation time.

The face recognition system determined the positions of localized features (eyes, etc.) which were part of a larger object. The input (image) is first processed on a coarse scale to reduce computation. Scanning localized feature detectors over an image provided translation invariance. Post-processing of feature detector output was used to handle false detection and provided some rotation invariance. In the gender recognition application, the NN performance was comparable to that of humans viewing higher resolution images. Humans and the NN tended to misclassify (male/female) the same (ambiguous) images.

CHAPTER V

SEEDLING IMAGE PROCESSING AND FEATURE EXTRACTION

Many NN applications benefit from reduced dimensionality of their inputs. These benefits are realized through reduced computation time, training time, and storage requirements. Although additional computation is incurred in generating the lower dimensional inputs, the net result should be reduced total computation. An argument against such preprocessing is that a complete understanding of the problem is required in order to know what features to extract from the original data. There is a danger that important features will not be extracted and presented to the network. On the other hand, successful networks are often developed using "obvious" features extracted from the original data, while poorer performance would be achieved using only the original data. This result occurs because, in training, the network may not effectively extract the obvious features.

Image processing operations providing dimensional reduction and feature extraction in support of root collar (RC) recognition are presented in this chapter. Image acquisition and characteristics of the seedling images used in this work are first discussed. Features extracted from seedling images for use by the heuristic algorithm and as inputs to neural networks are then described. Feature names, definitions, and computational formulas are summarized in Appendix A, Tables A1 and A2. Values of selected features, illustrating their relationship to the seedling stem region and root collar location, are plotted for five seedlings (Fig. 4) in Appendix B, Figures B1 through B19.

Seedling Image Acquisition

A silhouette, gray-level image is acquired by transporting a seedling between a camera and backlight. A 2048-pixel line-scan camera is used, fixing the spatial resolution along image lines. The camera typically has a 200-mm (8-in) field-of-view, corresponding to a 0.1-mm pixel resolution. Image lines are acquired synchronously with seedling displacement, typically at 1-mm intervals.

The large difference between horizontal and vertical resolution is intentional. High transverse resolution is desirable for precise measurement of stem diameter. High resolution is also required to image small diameter roots. Lower longitudinal resolution reduces the amount of image data, and thus, the computational burden, allowing greater seedling inspection rates. The measurement precision which can be obtained for longitudinal features such as shoot height and root length using 1-mm resolution is considered acceptable.

Thresholding

The backlit seedling image exhibits high contrast between seedling and background. Each pixel is digitized to an 8-bit gray scale in which 0 is black and 255 is white. Thresholding is used to segment the seedling from the background. The result is a

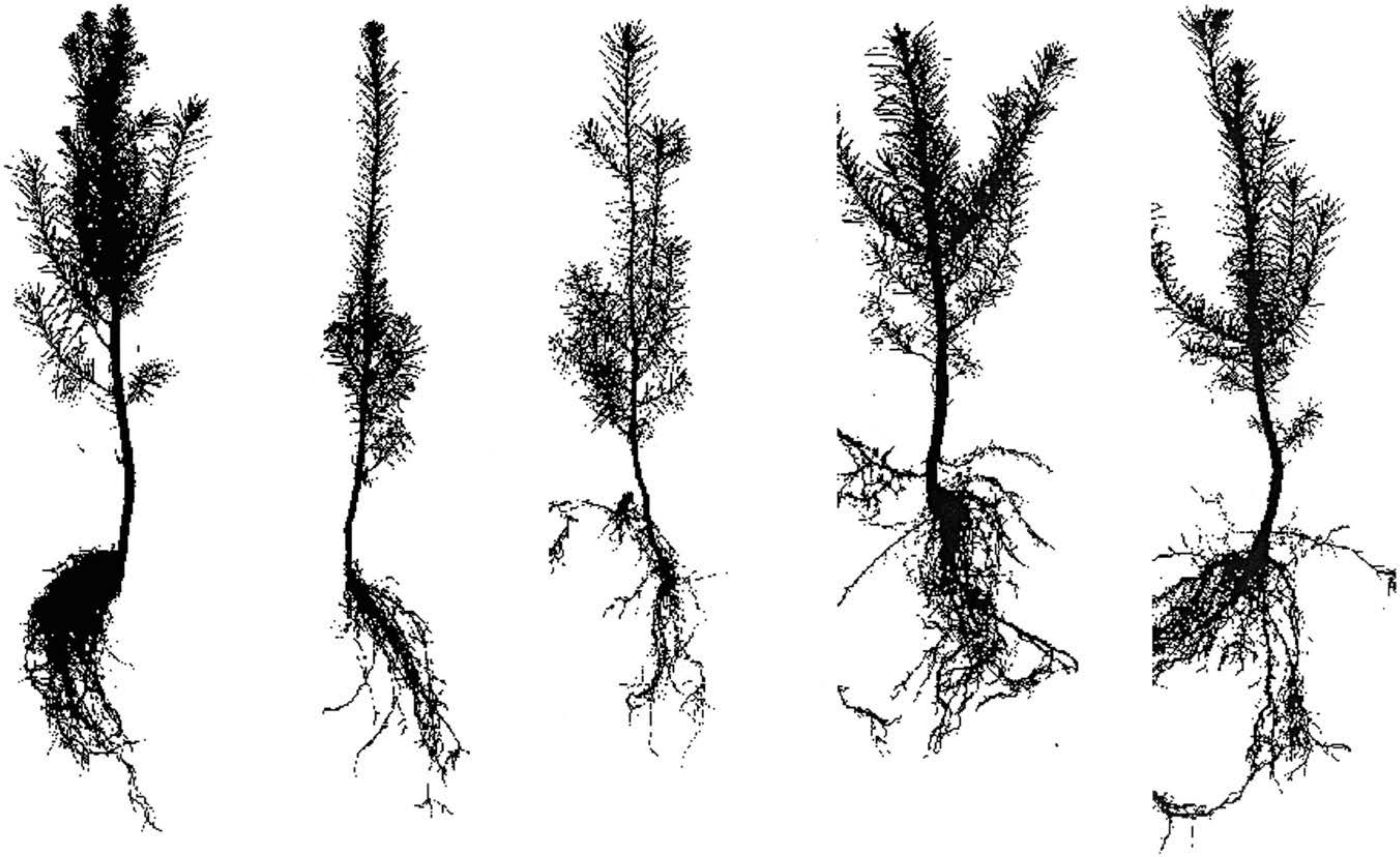


Figure 4. Douglas fir seedlings, DF001 - DF005.

binary image in which seedling pixels have an intensity of 1 and background pixels have an intensity of 0.

Seedling gray scale intensities are generally close to zero, but may be as high as 48. Brighter seedling pixels occur at the seedling boundaries where the "image" falling on an individual pixel is partially seedling and partially backlight. Similarly, the image of fine roots (<0.1 mm) will not completely "cover" a pixel, even if centered on the pixel, so these pixels have higher intensities. Additional factors which increase seedling pixel intensity are transmission of backlight through the seedling and reflection of ambient light from the seedling surface.

Background (backlight) gray scale intensities are generally not saturated (255) due to competing requirements. Short integration (exposure) time decreases background brightness, but is desirable to freeze seedling motion and image small roots (with longer exposure time, a small moving root contributes less to the integrated pixel exposure). Smaller lens apertures decrease background brightness, but are desirable for increased depth-of-field, which yields sharp image focus over a greater range of distances from the camera. Depth-of-field is important in this application, because seedlings resting on the conveyor surface may occupy a depth of 76 mm (3 in) or more.

The combined attributes and constraints described above yield an intensity distribution in which seedling pixels range from 0 to approximately 48, while the maximum background intensity is approximately 100. A threshold intensity value of 48 has been used to convert the gray-level image to a binary image, which, qualitatively provides a good representation of the seedling (Fig. 4). Selection of a higher or lower threshold would increase or decrease, respectively, the number of “seedling” pixels in the binary image.

Thresholding is performed in real time (as pixels are digitized) by circuits on the line-scan digitizer card (DT2856; Data Translation, 1990). Hardware implementation reduces computational demands on the CPU, enabling faster processing rates.

Run-Length Encoding

Seedling image data is transferred from the digitizer card to host computer RAM for image processing. The digitizer supports the transfer of several different data formats: original gray-level pixels, binary pixels, and run-length code. The gray-level format consists of 2048 bytes per line, transferred as 1024 16-bit words. Binary data is compressed such that 16 pixels occupy a 16-bit word. Each binary image line is transferred as 128 16-bit words. Run-length encoding offers further compression, depending on image complexity.

A binary image is composed of strings (runs) of 0’s and 1’s corresponding to background and seedling pixels, respectively. The transitions between these runs are extracted to generate the run-length code. The color of the run (0 or 1) and its beginning position (0 - 2047) are encoded in a 16-bit word. The first run on a line always begins at pixel 0, but may be of either color. A binary image may be completely reconstructed from its run-length code. For seedling images, the number of runs on a line is generally less than 64, but may be as high as 128 (8:1 compression). Lines which contain only the seedling stem (or a single root crossing) are composed of three runs: background, seedling, and background.

Run-length encoding thus provides significant data compression while completely preserving the binary image. The encoding process is implemented in real time on the line-scan digitizer board. Hardware implementation reduces the computational demands on the CPU. Compression results in reduced transfer time of image data from the digitizer to RAM. Further, and significantly, feature extraction can be achieved more efficiently (less computation) using run-length code rather than the original binary data.

Timing analysis indicates that run-length code transfer (through the ISA bus) is the rate limiting operation for seedling inspection. Thus, this data compression technique significantly increases the seedling inspection rate. Alternative image representations would require greater transfer and processing time.

Features Used by the Heuristic Algorithm

The heuristic algorithm uses several features which are extracted from the run-length code. These are computed for each line after the run-length code for each line has been transferred from the digitizer to RAM:

Runs - number of "seedling" runs (Fig. B1).

Area - sum of seedling run lengths (Fig. B5).

Span - distance from the beginning of the first seedling run to the end of the last seedling run (Fig. B9).

Filtered area - This feature attempts to recognize a single run on each line which could be the seedling stem near the root collar. The number of runs having a length within limits corresponding to the minimum and maximum expected seedling diameter is determined. If the number of such runs is not 1, filtered area is assigned a value of 0. Otherwise, the value of filtered area is computed as the average expected diameter, minus the absolute value of the difference between actual run width and average expected diameter.

As previously described, these features are extracted from each line in the image, multiplied by weighting factors, and summed, generating a value for each line in the image. After smoothing, this set of values may be interpreted as a function (signal) with many peaks and valleys. Through appropriate selection of weighting factors, the root collar will be located at a local minimum (valley) of the curve.

Features Used by Neural Networks

The feature, "Runs," was used as a neural network input. "Area" and "Span" were used only after transformation. "Filtered area," as described above, was not used. Several transformations of the features Runs, Area, and Span were generated and used in various networks. The simplest transformation used was inversion (1/Runs (Fig. B4), 1/Area (Fig. B6), 1/Span). This transformation effectively compressed the range of each feature (0-1), while expanding the percentage of the variable range associated with the root collar. For example; lines with 1, 2, and 3 runs yield 1/Runs feature values of 1, 0.5, and 0.33, respectively. Increasing numbers of runs yield smaller incremental changes in the feature value. Small numbers of runs (1-3) are of more interest (associated with the root collar) than large numbers of runs and are represented by a larger portion of the variable range using the inverse transformation.

The features described were extracted from each image line and thus had only local scope. Feature values can vary significantly between adjacent lines. Therefore, smoothed features were computed (SmRuns (Fig. B2), SmArea, SmSpan) by averaging the original feature values over a local neighborhood (11 lines). This widened the scope of the feature and reduced its local variation (noise). The trend of a feature value was also recognized as a potentially useful feature. Area, for example tends to decrease above the root collar and increase below the root collar. Therefore, derivatives of the smoothed features (DelSmRuns (Fig. B3), DelSmArea, DelSmSpan) were computed and used as additional features. The remainder of this section describes additional image processing and feature extraction techniques used to generate features for neural network root collar recognition.

Run-Length Histogram

Statistics of the set of runs on each line, such as mean and variance of run lengths might be computed. More descriptive, however, would be a histogram of run lengths. This histogram encodes the frequency distribution of run lengths.

Lines in the foliage region tend to have a large number of small runs due to needles, and a few long runs due to dense foliage. Lines near the root collar contain a single run with a length within the expected range of stem diameters, in addition to a few needle or root runs. Lines in the root zone have a large number of runs, but typically not the long runs associated with the foliage mass.

Since the camera has 2048 pixels, a large number of bins (2048) would be required for a full-resolution histogram. The run-length histogram need not have full resolution, nor uniform bin sizes. A couple of bins for short runs corresponding to needles and roots (<2.0 mm) should be adequate. A bin width of 1-2 mm over the expected range of seedling diameters (2-12 mm) would detect potential stem runs. A few more bins can be used to record runs longer than the maximum expected stem diameter (>12 mm). An 8-bin histogram (Hist1 - Hist8) was defined and used in this study. Two bins were used to encode short runs (0-1.0 mm, 1.0-2.0 mm). Four bins were used for run lengths corresponding to the expected range of seedling diameters (2.0-4.5 mm, 4.5-7.0 mm, 7.0-9.5 mm, 9.5-12.0 mm). These should detect potential stem or RC runs. Two bins were used to record long runs (12-24 mm, >24 mm). Five of the eight histogram bins are plotted for one seedling in Figure B12.

Moments and Moment Invariants

Moments comprise a large set of features which may be used for object classification or recognition. Moment invariants are functions of moments which are insensitive to object translation, scaling, and rotation. They can be used to identify unique shapes regardless of location, size, or orientation (Jain, 1989).

The 0th order moment is the area of the object and is invariant. Area is easily computed by summing the length of individual runs. Higher order moments, due to their non-linearity, may provide features which are more sensitive to the root collar location. The 2nd and 3rd order central moments were computed from the runs on each image line.

The 2nd and 3rd order central moments had relatively small values near the root collar and very large values in the foliage region. The natural logarithm of the absolute value of the 2nd and 3rd central moments (Figs. B10 and B11, respectively) was computed and used as a neural network input. The logarithm compressed the range of the feature values while giving small values (associated with the root collar) a larger percentage of the variable range.

Connectivity Analysis

Connectivity analysis is a powerful technique for segmenting objects in an image from each other and the background. Many useful features can be extracted from the resulting connected regions. Connectivity analysis typically begins by determining for each pixel which of its four or eight neighbors have the same value (4-connectivity or 8-connectivity)(Gonzalez and Woods, 1992).

Within-line connectivity analysis of seedling images is performed in the process of run-length encoding. Each run is a connected sequence of pixels. By definition, runs on the same line cannot be connected. Runs on two adjacent lines are connected if they share a mutual column index.

Connectivity analysis was used to segment the seedling stem from other seedling components (needles, branches, roots, and debris) on the same image line. A sequence of runs (one per line) which yields a connected path from the top (foliage) to the bottom (roots) of the image is desired. For most seedling images, the "stem region" containing the root collar is guaranteed to be contained within the connected run sequence (path). Exceptions occur when long roots or branches parallel the stem region and form a secondary connected path between the foliage and roots. Therefore, multiple connected run paths were tracked. These would often join and split from line to line. Tracking multiple paths generally precluded the algorithm from stopping at a "dead end" created by a downward-oriented branch. The number of paths was limited to reduce computation and storage requirements.

A simple algorithm was implemented to perform seedling connectivity analysis. Starting at the top of the image, "connected" runs were identified on a line-by-line basis. The analysis was initialized by selecting the longest run on the first image line as the initial connected run. All runs on the next line which overlapped the connected run(s) on current line were determined. The longest three runs (if more than 3) were recorded as connected. If no connected runs are identified on a given line, the algorithm is re-initialized with the single longest run on that line.

Many features were extracted from the connectivity analysis result for each line. These included the number (ConRuns, Fig. B13) and sum area (ConArea) of the connected runs. Within the stem region, seedling runs close to the stem (but not connected) may provide additional support for root collar identification. These runs correspond to needles, branches, roots, and loosely attached debris. Therefore, these runs were identified and labeled as "close connected runs." A run is defined as "close" if it is within 50 pixels (approximately 5 mm) of a connected run on the same line. The sum area of close connected runs (CloseArea, Fig. B17) on each line was used as feature for root collar recognition.

Further transformations were applied to the connectivity analysis result to produce alternate features. These included computing the average value (CloseA11) over a small window (11 lines) and clipping (CloseA11C50, Fig. B18) at some maximum value. One transformation was used extensively with various window sizes. In it, the minimum connected area within a large window (160 lines) was subtracted from the average (or minimum) connected area from a small window (5 to 20 lines; ConA11C50, Fig. B16). The minimum within a large window was also subtracted from the minimum within a small window (ConMn11C50). These transformations offset the connected area value such that the minimum (near the root collar) is always zero, rather than the stem diameter of each particular seedling (ConMn41, Fig. B15). From CloseArea, the clipped maximum value within a window was computed (CloseMx11C50, Fig. B19).

Composite Feature Sets

The large number of features used in this study were grouped into composite feature sets to reduce the total number of experimental treatments. Composite feature sets are defined in Appendix D, Tables D1 and D4.

Composite features sets were formed by grouping several related features. The eight histogram bins were grouped into the composite feature set, “h,” for example, and the two moment-based features were grouped into the feature set, “e” (Table D1). Feature sets “a,” “c,” and “d” were composed of the inverse, smoothed, and derivative transformations, respectively, of the features, Runs, Area, and Span. For some experimental treatments, the effect of an individual feature was of interest, so *that* feature was given a composite feature identifier, to be consistent. This procedure was followed for the features, Runs (“r”) and ConRuns (“f”), for example (Table D1).

One or more composite feature sets were used as network inputs in a given experimental treatment. All feature sets were not tested with all network architectural variations. Composite features used by specific neural networks are listed in Tables D2, D3, and D5 along with network architecture and root collar recognition performance.

CHAPTER VI

NEURAL NETWORK APPROACH FOR ROOT COLLAR RECOGNITION

The difficulty of locating the seedling root collar, both manually and with a machine vision system, has been discussed. The machine vision approach relies on the extraction and processing of features which are sensitive to seedling and root collar morphology. In the previous section, a large number of features were described, however, these are difficult to integrate into a root collar recognition algorithm. Yet, the good performance of the heuristic algorithm indicates that a solution may exist.

These concerns can be addressed through the use of a neural network. As previously described, processing elements (PE's) in a neural network apply weights to their inputs, sum these, and apply a non-linear transfer function to produce an output. The PE computation and architecture of a neural network allow the implementation of complex non-linear functions. Most importantly, training algorithms allow neural networks to learn a functional relationship between inputs and desired outputs through the presentation of training data.

Neural networks offer several advantages over the current heuristic algorithm. They naturally handle complex and abstract sets of input features (Wang, 1993). They implement a complex and non-linear functional relationship between these inputs and desired output (root collar location). Neural networks learn the functional relationship between input and output through presentation of training data, relieving an operator from the task of heuristic algorithm parameter adjustment and optimization.

The networks investigated are characterized by three different architectures; NN1, NN2, NN3. Among these networks, alternative input features, processing element configurations, and outputs were investigated. A total of 243 different networks and 57 input features were investigated. Individual networks used from 2 to 60 inputs and generated 1 or 4 outputs.

Network Inputs

There are many alternative input representations for this pattern recognition application. The whole seedling image could be input to a neural network. This approach would require a very large network and result in slow execution speed and long training times. Run-length code could be used as network input. This technique provides significant data compression, but the network may still be too large for real-time execution.

Further data compression may be achieved by extracting features from the run-length code. This is the approach used in this investigation. Initially, features used by the heuristic algorithm were investigated as network inputs. Subsets of these, and additional features were used to test and compare the performance of networks using different feature sets.

Given features extracted from each image line, a significant architectural issue is how to handle the variable number of lines (400-900) in a seedling image. A network could be defined to accommodate some maximum number of lines. For images with fewer than the maximum number of lines, the unused inputs must be set to appropriate values. Larger images might safely be cropped, since the root collar generally occurs in the central portion of the image.

The approach taken in this work is to develop networks which detect a localized feature (root collar) by using inputs sampled from a variable-sized window, or field-of-view (FOV) within the seedling image. This network may be evaluated at all (or a subset of) image lines and post processing of the network outputs performed to determine the RC location.

Network Output(s)

The networks investigated use inputs consisting of multiple features obtained from one or more image lines. Obviously, the network must generate one output indicating whether or not these inputs correspond with the root collar. The foliage, stem, and root regions of a seedling have qualitatively different morphology and a consistent spatial relationship. The root collar is located within the stem zone, below the foliage and above the roots. Networks trained to identify each of these regions, in addition to the root collar, might exploit the morphological differences and spatial relationship between them to achieve superior root collar recognition performance. Therefore, one-output (RC) and four-output (RC, Foliage, Stem, Roots) networks were investigated.

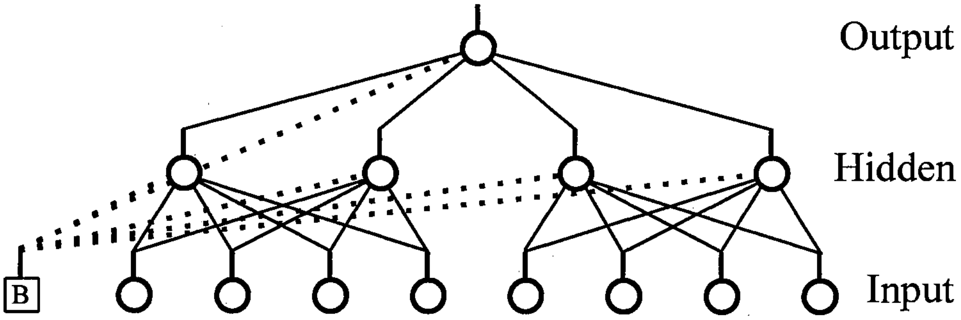
Network Architecture

Given input features and network outputs, many architectural alternatives are possible in the definition of a specific network. Based on the previous discussion, the number of inputs per line, the number of input lines, and the spatial relationship of those lines are obvious variables. These further define network input.

There are many configuration alternatives for the hidden layers of a network. These include the number of hidden layers, the number of PE's in each hidden layer, and the connectivity between PE's. Networks with too many hidden layer PE's may "over-learn" the training data and perform poorly with previously unseen inputs. Too few hidden layer PE's may inhibit the networks ability to learn a complex relationship. Fewer PE's are desirable for reduced computational burden. Therefore, for a given set of inputs and outputs, several networks with different hidden layer configurations were trained and tested.

Variations in PE connectivity were investigated to a lesser extent. The default connectivity is "fully connected." In a fully connected network, every PE in a given layer is connected to each PE in the following layer. Full connectivity was used for all NN1 and most NN2 type networks. Full connectivity was always used between the first and second hidden layers for networks which had two hidden layers. Full connectivity was always used between the last hidden layer and the output layer.

Two variations of partial connectivity were used for NN3 and several NN2 type networks. Line-connectivity brought all features from a limited set of lines together as inputs to a set of hidden layer PE's. Feature-connectivity brought a single feature from a larger set of lines together as inputs to a set of hidden layer PE's. These are illustrated in Figure 5. NN2 and NN3 networks had inputs from multiple image lines and thus a potentially large total number of inputs and associated connections. Multi-line inputs were spatially ordered from the top to the bottom of the seedling image. Partial connectivity was used to exploit the spatial relationship of these inputs. Partial connectivity reduced the total number of connections, and thus, the computational burden of network evaluation.



								Line Connectivity
1	2	1	2	1	2	1	2	Feature
1	1	2	2	3	3	4	4	Line

								Feature Connectivity
1	1	1	1	2	2	2	2	Feature
1	2	3	4	1	2	3	4	Line

Figure 5. Line and feature connectivity example for 4 input lines with 2 features per line and 2 hidden layer PE's per connected group.

NN1 networks

These networks accepted 2 to 9 inputs from a set of 27 features. Inputs were generally extracted from a single image line, and thus, have narrow scope (Fig. 6). As an exception, some features were averaged over a small set of lines (11) centered about the evaluated line. Derivative features were computed from feature values before and after the evaluated line. NN1 networks generate 1 or 4 outputs per line. One output classifies the line as RC (+1) or Not RC (-1). Additional outputs classify the line as Foliage, Stem, and Roots, or Not.

NN1 networks are the simplest of the networks investigated. They require the least amount of computation, and therefore allow fast evaluation. Their local scope allows evaluation line-by-line as image data become available.

NN2 networks

These networks operate as the second level in a hierarchical neural network. The first level is NN1, previously described. NN2 accepts 1 to 4 outputs from NN1 as inputs (Fig. 7). NN2 has a broad scope, because inputs are accepted from a FOV of 140 image lines (140 mm). Fifteen lines sampled 10 lines (10 mm) apart were used to cover the FOV, while limiting the number of inputs. Using 1 to 4 outputs from each image line of NN1, NN2 had 15, 30, 45, or 60 inputs. NN2 produced a single output which classified the line at the center of the input FOV as RC, or Not RC.

These networks have intermediate complexity among the three architectures investigated. They rely on output from NN1 which should be quickly computed for each image line. NN2 attempts to provide more robust root collar recognition by integrating information from a broader scope than NN1.

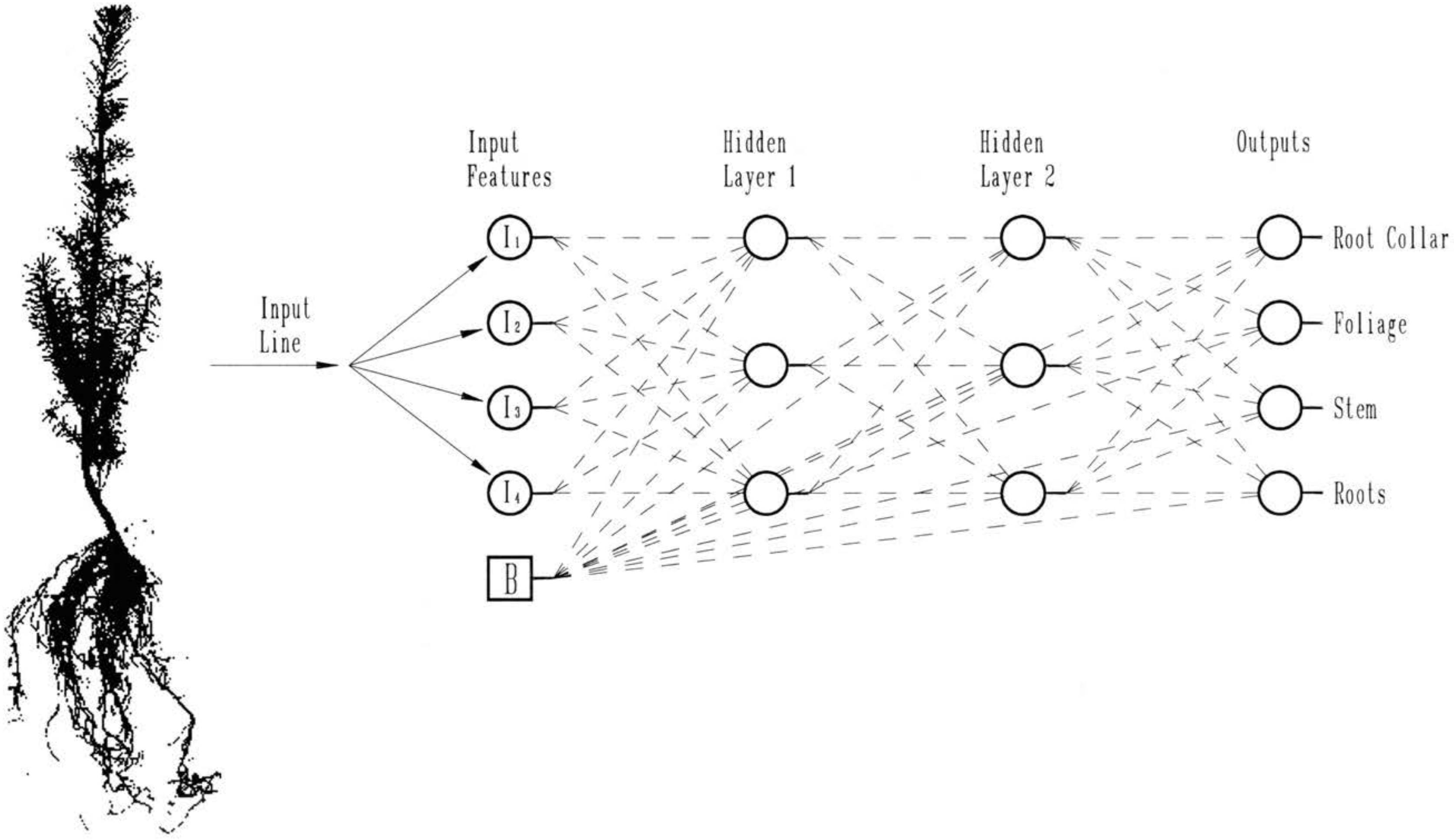


Figure 6. NN1 architecture example. Four features from a single line as inputs to a network with three PE's in each of two hidden layers and four outputs.

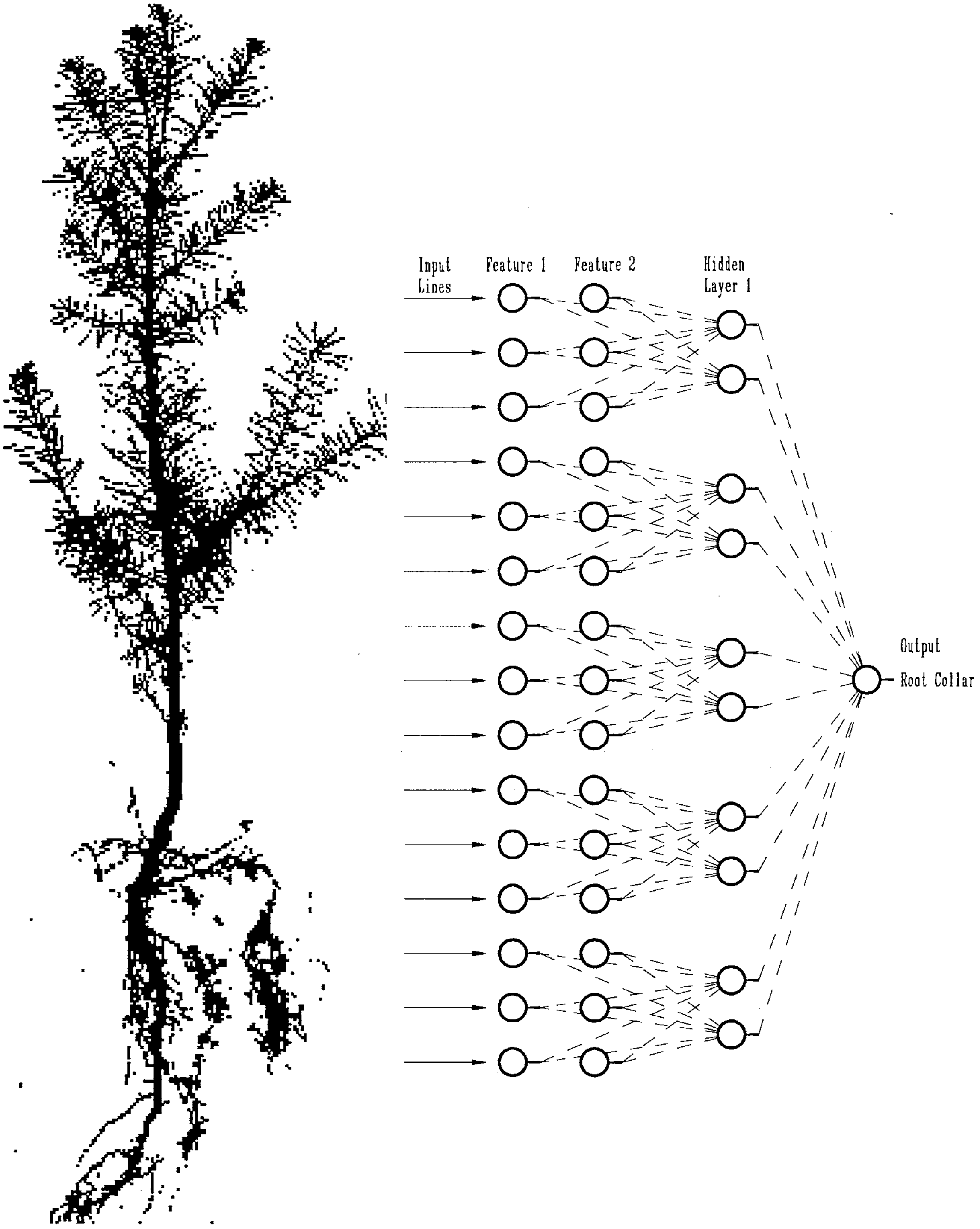


Figure 7. NN2 and NN3 architecture example. Two features from fifteen lines as inputs to a network with ten PE's in a single hidden layer and one output. Line-connectivity used between groups of three input lines and two hidden layer PE's. Bias PE not shown.

NN3 networks

These networks accepted image feature inputs (like NN1) over multiple image lines (like NN2). NN3 produces a single output classifying the line at the center of the evaluation FOV as RC or not RC (Fig. 7). Two groups of NN3 networks, characterized by wide (70-280 mm) and narrow (14-42 mm) FOV's were investigated. Input FOV size (14, 28, 42, 70, 80, 100, 140, 210, 280 mm), number of input lines (9, 15, 21), and number of inputs per line (2, 4) were varied along with different input features and network architectures. Different feature sets and architectures were used, in general, for the wide and narrow FOV groups.

These networks were the most complex investigated. While NN1 performed the same computation on each line in the image, NN3 (like NN2) is capable of performing different computation on the features from each line in the input FOV.

Data Acquisition

The seedling inspection system was used to acquire seedling images and store them in run-length code format. Software enhancements supporting interactive root collar identification were added to the seedling inspection system. These refinements allow the operator to use the mouse cursor to select an individual run in a seedling image viewed on the inspection system monitor. The selected run is highlighted in the seedling image. Through a menu selection, the selected run is defined as the “true” location of the root collar. Similarly, the top and bottom of the stem region are identified. The actual seedling was manually inspected to aid in the identification of the corresponding features in the seedling image.

Several image and seedling attributes are stored in the header of the run-length code file. These attributes include spatial calibrations and the heuristic algorithm root collar location. The manually identified feature locations were added to the run-length code file header.

An independent program was written to read the run-length code files and extract features for use by the neural networks. Results were written to a feature data file. Feature data files were further processed to compute alternative features (smooth, derivative, etc.) and to scale and format feature values for use in network training and testing. A commercial data transformation program, DataSculptor (NeuralWare, 1995b), was used for this purpose. Network training and test files were composed of records, each of which consisted of a single set of network inputs and desired output(s). Network training files were generated for each unique set of network input features.

Training and Test Data

Neural networks were trained from images of 50 Douglas fir seedlings. Network performance was evaluated using an independent set of 50 Douglas fir images. Performance of selected networks was investigated further using 100 additional Douglas fir and 150 ponderosa pine seedling images.

Statistics associated with training and test group images and manually identified seedling features are presented in Table 3. Images for the training group "DF1-50" and test group "DF51-100" were acquired at the same time. These seedlings had been root pruned at a commercial nursery. DF51-100 seedlings were shorter (image height, Table 3) but similar to the training seedlings with respect to other morphological features. DF101-200 seedlings came from a different seedlot and had not been root pruned. Average normalized RC location is much closer to the top of the image, and the stem region is shorter for these seedlings than for the other Douglas fir seedlings. Ponderosa pine seedlings (PP1-150) differed from Douglas fir by having longer needles, a higher normalized RC location, and an RC location closer to the top of the stem region.

Table 3. Image and morphological statistics for training and test seedlings

Morphological Feature		Train DF1-50	Test DF51-100	 DF101-200	 PP1-150
Image height, mm	Avg	711	619	667	628
	Stdev	146	139	91	116
Normalized RC location in image, %	Avg	60	59	50	45
	Stdev	5	7	5	5
Height of stem region, mm	Avg	51	45	37	48
	Stdev	18	16	10	15
Normalized RC loc. within stem region, %	Avg	33	29	29	11
	Stdev	19	17	12	11

Network training utilized only a fraction of the data from the training seedling images. Recall that network outputs are associated with a single image line. Inputs are obtained from the same image line or multiple lines (up to 21) sampled from a variably sized FOV (14 to 280 mm). Only one line of 700 (average image height) is defined as the root collar location, a very small percentage. Thus, relatively few RC examples were available for network training, compared with the abundance of Not RC examples.

The root collar location was therefore "fuzzified" to increase the number of root collar training examples. Lines near the root collar were assumed to be similar to the root collar. A trapezoidal fuzzy membership function was used to define additional root collar lines. The top and base of the trapezoid were nominally 3 and 11 lines wide, respectively. The base of the trapezoid was clipped when it extended beyond the stem region. The fuzzy membership function classified the two lines adjacent to the root collar as RC and further lines as RC with decreasing confidence. Thus, the number of root collar examples was increased to 9 of 700 lines, still a small percentage. Equal numbers of RC and Not RC examples were desired for network training. Therefore, all of the fuzzified root collar lines were used, and an approximately equal number of the remaining image lines were sampled uniformly for use in network training. Thus, approximately 1,000 of 35,000 lines (3%) from 50 images were used for training. All lines were used for network testing, including tests performed during training to assess network performance and convergence. A limited test was performed to compare networks trained using rectangular and trapezoidal fuzzy membership functions.

Neural Network Development and Training

A commercial neural network development package, NeuralWorks Professional II Plus (NeuralWare, 1995a), was used for network definition, training, and testing. Multilayer feedforward networks trained with the backpropagation algorithm were used, as described in Chapter IV. Equations 1 and 4 were used for feedforward computation.

Equations 8, 12, 13, and 15 were used for training. Learning rate, η, was set to 0.075, 0.0625, and 0.0375 for PE's in the first hidden layer, second hidden layer, and output layer, respectively. The momentum coefficient, α, was 0.4 for the hidden and output layers. Transfer function offset, β, was 0.1. Weight changes were accumulated for 16 training examples before weight updates (epoch size). Epoch size was approximately equal to the average number of training examples per seedling. Training examples (input/output data record) were presented in random order.

Training proceeded until network convergence, or 100,000 epochs, whichever came first. During training, network performance was evaluated every 1,000 epochs to determine network convergence status. Performance was measured by an objective function evaluated using all lines from the training images. Convergence was declared when performance improvement was not observed after 20 test evaluations (20,000 epochs). Networks weights were saved when performance exceeded the previously measured best performance. For each network configuration, four different networks were trained by initializing each with different random initial weights. Trained networks were tested using all lines from the test images.

Objective function

An objective function was used to evaluate network performance periodically during training. Classification rate was selected as the objective function after evaluating several alternatives. Classification rate is the average correct classification percentage for all outputs. Network outputs range from -1 to +1 (hyperbolic tangent activation function). An RC PE output greater than 0 was classified as RC while an output less than 0 was classified as Not RC. The same procedure was followed for other outputs (Foliage, Stem, Roots) and desired outputs. Output classification was compared with that of the desired output to compute classification rate.

CHAPTER VII

PERFORMANCE EVALUATION

An independent program was written to post-process the NN output from all lines in a seedling image and select a single line as the root collar location. Response examples for selected best performing networks are presented in Appendix C, Figures C1 through C8. Some NN1 networks (not shown) exhibited large variation between adjacent lines. For some seedlings, multiple regions within the image generated significant RC responses. The RC output was smoothed (11-line average) to reduce local variation. For each seedling, the maximum and minimum RC response was determined and a threshold set at 75 % of that range. The midpoint of the widest response peak exceeding the threshold was defined as the network's RC location.

Root collar location performance was primarily quantified and assessed by the mean and standard deviation of the root collar location error. Root collar location error is the difference between the manually identified and the neural network identified RC locations. Minimum mean error and error standard deviation are desired.

Two supplemental measures were developed to further assess NN performance. Percent "on stem" is the frequency with which the NN RC location was found within the bounds of the manually identified stem region. Diameter and other morphological measurements should be more accurate if the RC is located within the stem region.

Percent "found stem" is the frequency with which the NN RC response overlapped the stem region. This statistic indicates how often relatively large errors occurred. Data were not available to compute the "found stem" statistic for the heuristic algorithm.

For each configuration of input features and NN architecture, four different networks were trained by initializing each with different random initial weights. Performance measures (RC location error mean and standard deviation, percent "on stem," and percent "found stem") were averaged for the four networks. These results are reported in Appendix D, Tables D2, D3, and D5. Test seedling RC location error for the four repeated networks was statistically analyzed using the F-test and paired-sample t-test. Mean RC location error and error variance among the four networks were frequently statistically different at the 10% significance level. Therefore, the network yielding the best performance was identified (Tables D2, D3, and D5) and used for statistical comparison of networks having different configurations. The best network was defined as that network with minimum mean error and/or error standard deviation. When little difference between error statistics occurred, the "on stem" and "found stem" performance measures were evaluated for selection of the best network.

Statistical Performance Analysis

RC location error of the "best" networks was compared using the F-test and the paired sample t-test. Differences for all comparisons were considered to be significant at the 10% confidence level. Tests which exceeded this confidence level are printed in bold type. Statistical analysis was performed to compare input features and network architecture (number of PE's, hidden layers, connectivity, FOV). These results are presented in Appendix E, Tables E1 through E16 and discussed below. In each table, networks are generally ordered from left-to-right by average error. Since average error could be negative, the left-most entry is not always the best performing network. Therefore, the minimum mean error and minimum error standard deviation are printed in bold type. For two networks having similar performance, the one with lower computational demands is considered superior.

Statistical Comparison of NN1 Networks

One or four outputs

Networks with one output (RC) and four outputs (RC, Foliage, Stem, Roots) are compared in Table E1. Neither mean error nor error standard deviation differed significantly for the two network types, in general. Although not statistically significant (due to large variance) the four-output networks yielded the lowest mean error and error variance. For the one significant comparison, the four-output network had lower error variance. The bulk of NN1 networks were developed as four-output nets, which support the investigation of NN2 networks.

Objective function comparison

Several alternative objective functions were evaluated to determine which yielded networks with the best performance. In addition, the alternatives of a trapezoidal or rectangular fuzzy membership function (which defined the known root collar location for training) were investigated.

The default objective function (for NeuralWorks) is the rms error between the desired and actual network outputs. The classification rate objective function was previously discussed.

The confusion matrix integrates the desired and actual output for a given output PE. In contrast with the other objective functions, use of the confusion matrix allowed a single output (RC, for example) to be used to guide network training. The correlation coefficient of the confusion matrix was used as the objective function. The RC output confusion matrix was used to train networks in which the desired RC location was defined by trapezoidal and rectangular fuzzy membership functions. The "stem region" looks much like the RC, but is a much larger region, thus providing more training examples per seedling. A network which reliably identifies the stem region could be quite useful (more so than one which un-reliably identifies the RC). Therefore, RC location performance of a network trained using the confusion matrix for the stem output was investigated.

Results for five different neural networks (each the best of four repetitions) are compared in Table E2. Classification rate yielded the lowest mean error (though only significantly lower than the rms objective function) of the objective functions tested. Classification rate provided significantly lower error variation than most of the other objective functions. The rms error objective function (default) yielded the worst performance. Classification rate was used as the objective function for all subsequent network training.

Input features

Many different features were used as inputs to NN1 networks with several different architectures. Features were grouped into composite feature sets to reduce the total number of experimental treatments. Composite features used with NN1 networks are summarized in Table D1. Individual features are defined in Appendix A. All feature sets were not tested with all architectures. Performance of various feature sets is compared among NN1 networks with similar architecture in Tables E3a through E3e.

Among 14 networks with a single hidden layer containing 2 PE's (Table E3a), the one using feature set, "nae," provided the best performance. This feature set was comprised of six features; Nline, InvRuns, InvArea, InvSpan, M2, and M3. Performance of this network was superior to all but the next best network, using feature set, "nh." Eight networks incorporating the global feature, "Nline," generally yielded performance superior to that of the 6 networks not using the feature. The feature set, "nr," composed of only two features, "Nline" and "InvRuns," had the third best performance. Error variance among the six best networks did not differ significantly, but this group was generally superior to the remaining networks.

Among 18 networks with 3 PE's in a single hidden layer (Tables E3b and E3c), the one using feature set, "nru," composed of Nline, InvRuns, and NCArea (the only other global feature) provided the best performance. The best two feature sets (nru, nae) yielded significantly lower error than all but the third best feature set (na). Again, networks using the feature Nline were generally superior to those that did not. Error variance among the seven best networks did not differ significantly, but, in general, these had significantly lower variation than the remaining networks.

Among 11 networks with 4 PE's in a single hidden layer (Table E3d), the best four networks were generally superior to the remaining networks. The feature set, "nae," yielded poor performance with this architecture, but very good performance with all other architectures. Significant performance differences also occurred among 8 networks with 3 PE's in each of two hidden layers, and 5 networks with 4 PE's in each of two hidden layers (Table E3e).

Several observations can be made about the features investigated. Feature sets including "Nline" consistently yielded better performance than those that did not. Feature sets including "e" (moments M2, M3) performed better than similar feature sets that did not. Feature sets including InvRuns ("a") performed better than similar feature sets including Runs ("b"). Feature sets including "a" (InvRuns, InvArea, InvSpan) performed better than similar feature sets using the smoothed features, "c," (SmRuns, SmArea, SmSpan). Evidence more weakly suggests that feature sets incorporating derivative features "d" yielded better performance than similar feature sets that did not. Among the five architectures, the feature set, "nae," was best or second best, with one exception. Surprisingly good performance was obtained from the feature sets, "nr" and "nru," composed of 2 and 3 features, respectively. It was not expected that networks with so few input features would yield performance superior to others with the same and additional inputs. Thus, these feature sets were investigated in only one architecture each.

NN1 architecture

Several architectural variations were investigated by varying the number of hidden layers (1 or 2) and the number of PE's in each hidden layer. Statistical comparisons were performed among treatments with common input features. All architectures were not investigated with every feature set. Results are summarized in Tables E4a through E4d. Within these tables, features sets are ranked in order of best performance by the best architecture, while architectures are ordered by increasing number of PE's.

Statistically significant performance differences between architectures were observed within 7 of the 17 feature sets. For 5 of 10 feature sets tested with 3 or more architectures, significant performance differences were observed. Each of the five different architectures, however, yielded performance superior to some other architectures, among the various feature sets.

All architectures were tested with the best feature set, "nae." The two best architectures were those with the fewest PE's, a single hidden layer with three and two PE's, respectively. Performance of these two networks was not significantly different, but both were significantly better than the remaining three architectures. Error variation was comparable among 4 of the 5 architectures, but significantly larger in the architecture with 4 PE's in a single hidden layer.

Two- and three-PE architectures provided the lowest error for 11 of the 17 feature sets. Excluding cases where the 2- and 3-PE architectures were the only ones tested, they provided the lowest error for 7 of 13 feature sets. Considering only statistically significant comparisons, 2- and 3-PE architectures most frequently provided the lowest error. The 4-PE architecture was frequently superior to others when feature sets tested with only 2- and 3-PE architectures were not considered. Single hidden-layer architectures more often yielded performance superior to two hidden-layer architectures. Considering only statistically significant comparisons, single hidden-layer networks yielded superior performance in 9 of 12 cases.

Statistical Comparison of NN2 Networks

Input features

NN2 input features consisted of one or more outputs from a four-output NN1 network. The NN1 outputs corresponded to the root collar "C," foliage "F," stem region "S," and roots "R." Four feature sets were defined for input to NN2 networks; C, CF, CFR, and CFSR. These feature sets are compared among NN2 networks with common architecture and which were trained/tested with outputs from a common NN1 network in Tables E5a and E5b. The feature set, CF, yielded the lowest mean error in 4 of 6 cases. Considering only significant comparisons, CFR provided the lower mean error than another feature set in 4 of 8 cases. Error variation did not differ significantly among the feature sets.

NN1 network features are compared for treatments with common NN2 features and architecture in Table E6. The NN1 feature set, "na," yielded the lowest mean error in 3 of 4 cases, and significantly lower error than "nae" in 2 of 3 cases where they were compared. The feature set, "nae," yielded the lowest error variation in 3 of 4 cases, but this variation was not significantly less than that of "na." Thus, the best NN2 performance was achieved using inputs from an NN1 network that was not identified as the best NN1 network.

Network architecture

Network architectures are compared for treatments with common NN1 and NN2 feature sets in Tables E7a and E7b. Few significant differences between network architectures were observed. For the two comparisons which included a 15-PE architecture, lower mean error was provided by some treatments with fewer PE's. Five-PE architectures provided lower mean error in 5 of 7 comparisons. Five-PE architectures yielded lower mean error than another treatment in 2 of 4 statistically significant comparisons. The three reduced-connectivity treatments tended to provide lower mean error than the full-connectivity treatments. Therefore, networks with fewer PE's and reduced connectivity are considered superior due to lower computational burden.

Another architectural consideration is whether the hierarchical NN2 networks provided better performance than the NN1 networks alone. Performance of NN2 networks is compared with that of the corresponding NN1 network in Table E8. In the majority of cases, NN2 networks were superior to the NN1 network providing their input. In 14 of 23 cases, the NN2 networks had significantly lower mean error and error variation. In only one comparison did the NN2 network yield statistically inferior performance.

Statistical Comparison of NN3 Networks

Two groups of NN3 networks, characterized by wide (70-280 mm) and narrow (14-42 mm) FOV's were investigated. Different feature sets and architectures were used, in general, for these two groups. The two groups are therefore discussed separately and comparisons performed among different treatments within the separate groups. Subsequently, the wide and narrow FOV groups are compared using treatments which are most similar between them.

Performance of NN3 networks, in general, was very good, and therefore few statistically significant differences were observed between treatments.

NN3 wide FOV features

The 140-mm FOV was the most extensively tested of the wide FOV networks. Feature sets are compared among networks with similar architecture and a 140-mm FOV in Tables E9a and E9b. The few statistically significant differences observed occurred between the lowest performance treatment(s) and the best. Feature set "a2d0" and "m2d0" most frequently had the best performance for one and two hidden-layer networks, respectively. It appears that the feature derivatives (d0) were beneficial. The features, "a0," used only in single hidden-layer networks, had (statistically significant) greater error variation than the other features. This result indicates that clipped feature values (all other features) yielded superior performance. Feature "m5" yielded the largest mean error in 5 of 6 cases. In addition, feature "a5" tended to have a larger error variation than the average (excluding “a0”). Therefore, features which were clipped at the lowest magnitude (“a2” and “m2”) appear to be superior to the alternatives.

NN3 wide FOV architecture

Network architecture is compared for treatments with common FOV and feature sets in Tables E10a through E10e. Architectures are labeled “A” through “F” in order of increasing complexity. Architectures A - C have a single hidden layer, while D - F have two hidden layers. Significant differences generally occurred only between the best and worst cases in a test. Architecture D most often yielded the minimum average error, while A most often had the minimum error variation. Considering only cases where the best network was significantly better than other networks, architecture D had the lowest mean error in 5 of 10 cases and the lowest error variance in 7 of 12 cases.

NN3 wide FOV lines input

Most NN3 networks had 15 lines of input features with 2 features per line. Several networks were trained and tested with 9 or 21 lines of input. These are compared with 15-line networks having the same input features, and the most similar FOV in Table E11. Comparing 21 and 15 input-line networks, 15-line networks had lower error (none significant) in 6 of 8 cases. Fifteen- and 21-line networks had minimum error variance an equal number of times, but the 15-line network had significantly lower variation in one case. Differences were more dramatic comparing 15- and 9-line networks. The 9-line networks had lower mean error in 6 of 8 cases; significantly lower in 2 of 3 cases. Nine-line networks had lower error variance in 7 of 8 cases, and 5 of these were significant. This result is strong evidence that the networks with fewer input lines (among those tested) provided superior performance.

NN3 wide FOV comparison

The majority of wide FOV networks had a 140-mm field of view. These are compared with alternate FOV's of 70, 210, and 280 mm in Tables E12a and E12b for treatments with the same input features and architecture. The 210-mm FOV networks most frequently yielded the lowest mean error, and significantly lower error than other networks in 4 of 5 cases. The 280-mm FOV networks most often yielded the lowest error variance, and in 3 of 6 cases had significantly lower variance than other networks. Therefore, the larger FOV's appear to yield superior performance among the wide FOV networks.

NN3 narrow FOV features

Features are compared among narrow FOV NN3 networks having the same FOV and similar architecture in Tables E13a through E13c. No single feature set appeared to be superior across all FOV's. Differences between networks were generally not significant. Feature sets "cd," "bc," and "abc" most frequently yielded the lowest mean error for the 15-, 29-, and 43-mm FOV's, respectively. Feature set, "abc," most frequently yielded the lowest error variance. Feature set, "ce," yielded significantly larger error variance than the other feature sets. It differed from feature set, "cd," in having a clipped magnitude for one of its component features.

NN3 narrow FOV architecture

Network architectures are compared among treatments having the same input features and FOV in Tables E14a through E14c. Architectures are designated by the characters c, d, e, i, j, k (single hidden layer) and f, g, h, l, m, n (two hidden layers) in the tables. Not all architectures were tested with all features. The lowest error networks were again significantly different from only the highest error networks. Architecture, "f," the simplest two hidden-layer network most frequently provided the best performance. In 11 of 20 cases it yielded significantly lower error than other networks. Significant differences between error variance occurred with less frequency. Architecture, "i," with a single hidden layer, had significantly lower error variance than other networks in 7 of 9 cases.

Two connectivity variations may also be compared. In architectures c,d,e, two input features from a given line were connected to separate hidden-layer PE's (feature connectivity, Fig. 5), while in architectures i,j,k, the two features were connected to the same hidden-layer PE's (line connectivity, Fig. 5). Architectures c,d,e tended to have lower mean error, while architectures i,j,k tended to have lower error variation.

NN3 narrow FOV comparison

Narrow field-of-view networks were tested with 15-, 29-, and 43-mm FOV's. These are compared for treatments with common input features and architecture in Tables E15a and E15b. The 29-mm FOV most frequently yielded the lowest mean error, and this was significant in 12 of 21 cases. The 15-mm FOV had the lowest mean error in the remaining significant cases. The 43-mm FOV most frequently had the lowest error variation, but there were no significant differences.

Comparison of wide and narrow FOV NN3 networks

Narrow FOV networks are compared with a wide FOV network in Table E16. Although features and architectures do not match exactly, networks with the greatest similarity are compared. Multiple narrow FOV networks are included in some comparisons due to their similarity in features and architecture. A narrow FOV network provided lower mean error in 5 of 6 cases. None of these differences was significant. A narrow FOV network yielded lower error variance in 5 of 6 cases, and four of these were significant.

Summary of Network Performance

Among NN1 networks, a few feature sets provided significantly better performance than the rest of the feature sets investigated. All NN1 networks had relatively small numbers of inputs. The global feature, "Nline," significantly improved the performance of these networks which otherwise used features providing local information. The best feature set was “nae,” composed of 6 features. Good performance was achieved by two feature sets composed of just 2 and 3 features (nr and nru). Networks with two or three PE’s in a single hidden layer yielded performance superior to networks with more PE’s and hidden layers.

NN2 networks used the outputs from NN1 networks as input features. The feature set CFR (RC, Foliage, Roots; total of 45 inputs) yielded the best performance of the alternatives investigated. Networks with relatively few PE’s (3 or 5) and reduced connectivity provided good performance. NN2 networks were, in general, significantly better than the corresponding NN1 networks.

NN3 networks, as a group, provided the best performance of the networks investigated. Error rates among these networks were low enough that fewer significant differences were observed. Input features were based on connectivity analysis. Clipped feature magnitudes and feature derivatives appeared to be beneficial. The two hidden-layer architecture with the fewest PE’s provided the best performance. Fewer input lines (9 rather than 15 or 21) provided the best performance. Among both wide and narrow FOV groups, narrower FOV’s yielded lower mean error, while wider FOV’s yielded lower error variation. Comparing the two groups, narrow FOV’s (15- to 43-mm, rather than 150-mm or greater) provided superior performance. Among the narrow FOV networks, the 29-mm FOV provided the best performance.

Extended Testing of Best Networks

Network training and testing was performed using two groups of 50 Douglas fir seedlings (DF1-50 and DF51-100, respectively) which had been root-pruned at the nursery prior to use in this investigation. Several networks which yielded superior performance with respect to others of their type (five NN1's, one NN2, and four NN3's) were tested on two additional groups of seedlings. The first group consisted of 100 Douglas fir seedlings (DF101-200) which had not been root-pruned. The second group consisted of 150 Ponderosa pine seedlings (PP1-150) which had been root pruned. Morphological differences between these seedling groups were previously discussed and are summarized in Table 3.

Performance of the heuristic algorithm and the 10 selected networks is summarized in Table 4 and Figures 8 - 11. Average RC location error for 10 networks is presented in Figure 8. Results for the training seedlings and three groups of test seedlings are shown by adjacent bars. All NN's generally had lower mean RC location error than the heuristic algorithm for the training seedlings and DF51-100 test seedlings. The heuristic algorithm RC was above and the NN RC was below the manual RC location, on average. NN3 networks had lower mean error than NN1's and NN2 for these two seedling groups.

Table 4. Neural network* and heuristic algorithm RC location performance

Network		Train DF1-50				Test DF51-100				Test DF101-200				Test PP1-150			
Type	Name	RC error	Error stdev	RC on stem	Found stem	RC error	Error stdev	RC on stem	Found stem	RC error	Error stdev	RC on stem	Found stem	RC error	Error stdev	RC on stem	Found stem
		mm	mm	%	%	mm	mm	%	%	mm	mm	%	%	mm	mm	%	%
Heuristic algor.		-17	26	68	-	-7	21	70	-	-6	12	73	-	3	20	91	-
NN1	L4a	-2	18	84	97	6	20	75	96	31	72	80	84	69	80	54	67
NN1	L4h	0	13	90	97	4	17	84	97	19	43	86	89	56	71	75	77
NN1	L5e2	-1	13	89	97	5	16	92	96	29	51	78	81	62	75	73	75
NN1	L5f2	-1	15	89	98	5	16	90	98	39	59	71	74	68	73	67	70
NN1	L7e	3	16	89	99	7	21	79	93	23	43	84	89	72	71	54	67
NN2	H5c	4	16	90	98	5	16	80	98	29	62	78	88	70	87	63	77
NN3	M4i	-1	11	90	100	2	12	93	99	3	11	92	97	22	11	95	98
NN3	M8i	0	14	97	97	-1	12	95	96	4	11	96	96	20	12	95	96
NN3	W15	1	10	97	100	1	10	92	97	0	11	93	97	20	15	93	94
NN3	W7	2	11	97	99	2	13	92	97	-2	12	86	95	19	16	89	94

* average performance of 4 networks trained with differenent initial random weights

RC error = NN RC location - manual RC location

RC on stem = percent NN RC location within manually defined stem region

Found stem = percent NN RC response touches manually defined stem region

NN1's and NN2 had significantly greater mean error for test seedlings DF101-200 and PP1-150. These networks did not perform well on seedlings with morphology different from that of the training set; they generalized poorly. NN3's performed well on seedlings DF101-200, and had lower mean error than the heuristic algorithm. For seedlings PP1-150, however, which differed most from the training seedlings, NN3's had a larger mean error than the heuristic algorithm. One morphological parameter which may be a factor here is the normalized RC location within the stem region (Table 3). This parameter averaged 33% for the Douglas fir training seedlings and 11% for the Ponderosa pine seedlings, a 66% difference.

RC error standard deviation is presented in Figure 9. All NN's generally had lower error standard deviation than the heuristic algorithm for training and DF51-100 seedlings. NN1's and NN2 had significantly greater error standard deviation for DF100 and PP150 groups, which again indicates poor generalization. NN3's, however, had performance superior or comparable to the heuristic algorithm for these seedlings. The NN3 networks maintained good performance on seedlings with significantly different morphology than that of seedlings used for training.

Figure 8. Neural Network Performance - RC Location Error

Average RC Location Error of 10 Networks and Heuristic Algorithm

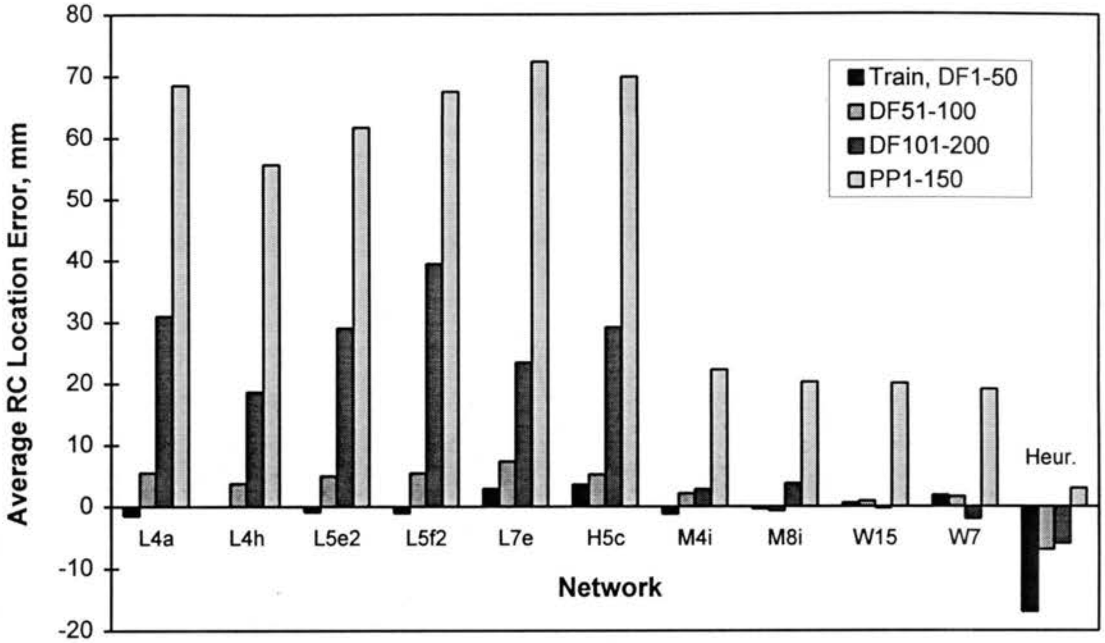
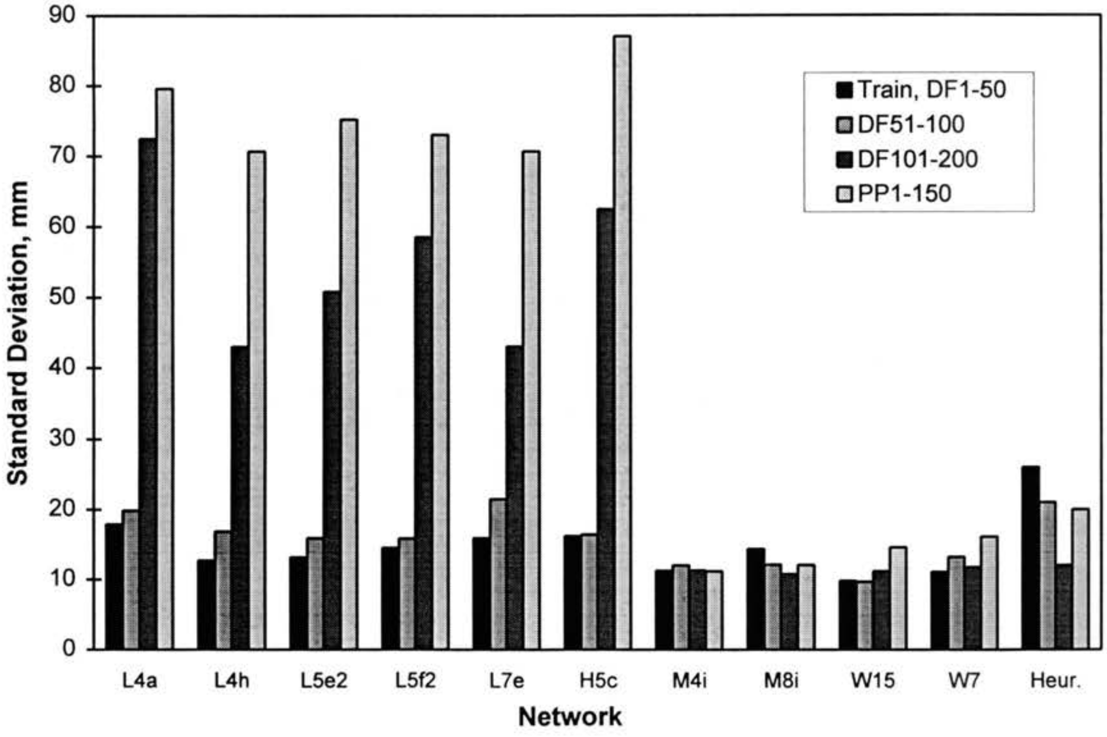


Figure 9. Network Performance - RC Error Standard Deviation

RC Location Error Standard Deviation of 10 Networks and Heuristic Algorithm



Neural network performance is further illustrated by the statistics "on stem" and "found stem," shown in Figures 10 and 11, respectively. Percent "on stem" is the frequency at which the RC was located within the bounds of the manually identified stem region. NN1's and NN2 were superior to the heuristic algorithm on this basis for the training, DF51-100, and DF101-200 seedlings. As previously indicated, however, NN1 and NN2 performance was exceeded by that of the heuristic algorithm for the Ponderosa pine seedlings. NN3's yielded the highest "on stem" performance. This performance level was maintained, for the most part, for all seedling groups tested. NN3 performance exceeded that of the heuristic algorithm for Douglas fir seedlings, and was comparable for Ponderosa pine seedlings.

Percent "found stem" (Fig. 11) is the frequency at which the NN RC response overlapped the stem region. This statistic was not available for the heuristic algorithm. When the RC location was not within the stem region, NN1, NN2, and NN3 found the manually identified stem region on an additional 7%, 13%, and 4% of the seedlings, on average, respectively. High performance levels for neural network stem region recognition represent a significant improvement over that of the heuristic algorithm. Once the stem is identified, current algorithms can be used to delineate it's extent and measure stem diameter.

Figure 10. Neural Network Performance - RC On Stem

RC Located within Manually Identified Stem Limits

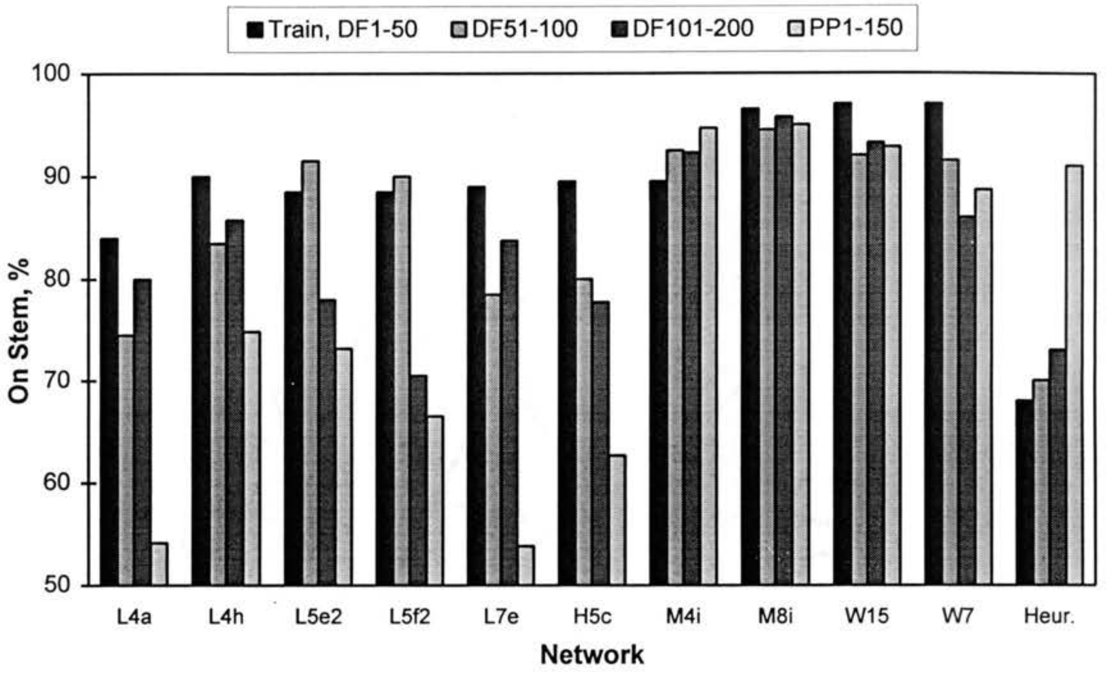
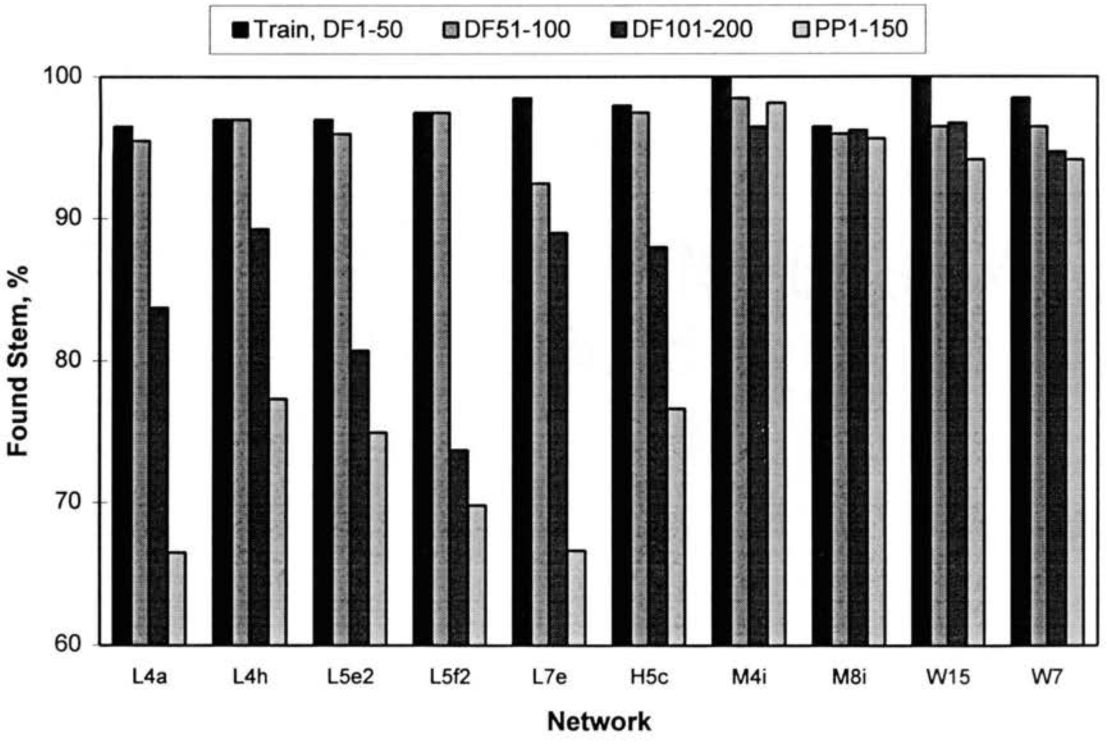


Figure 11. Neural Network Performance - Found Stem

NN RC Response Overlaps Manually Identified Stem Region



CHAPTER VIII

SUMMARY AND CONCLUSIONS

Neural networks which identified the root collar location in digital images of conifer seedlings were developed and found to provide performance superior to that of a heuristic algorithm. Performance was quantified by the mean root collar location error and the error standard deviation, with respect to manually identified root collar locations. Additional performance measures, "on stem" and "found stem," assessed the frequency with which the network identified the root collar location within the manually defined stem region and the frequency with which the network response overlapped the stem region, respectively.

The networks investigated were characterized by three different architectures; NN1, NN2, and NN3. Among these networks, alternative input features and network architectures were investigated. The networks accepted multiple input features extracted from a run-length encoded representation of a conifer seedling image.

NN1 networks were the simplest of the three groups. They used 2 to 9 inputs from a set of 27 features. Inputs came from a single image line (local inputs). Four outputs were produced, corresponding to root collar, foliage, stem, and root recognition. Significant performance differences between different feature sets and network architectures were observed. Invariant moment-based features, inverted features, and to a lesser extent, derivative features provided superior performance. Smoothed features were not helpful. A global feature, normalized line number, had a significant positive effect on NN1 performance. The best NN1 network used 6 input features. Good performance was provided by two networks which had only 2 and 3 inputs. Best performance was provided by the simplest networks, with 2 or 3 processing elements (PE's) in a single hidden layer. Performance of the best NN1 networks exceeded that of the heuristic algorithm for seedlings with morphology similar to that of the training seedlings. Network performance was poor, however, for seedlings with significantly different morphology.

NN2 networks functioned as the second level of a hierarchical network. NN2 inputs consisted of 1 to 4 NN1 outputs from each of 15 image lines, sampled over a 140-mm field-of-view (FOV). Fewer statistically significant differences between NN2 treatments were observed. Best performance was achieved using NN1 Root Collar, Foliage, and Root outputs as NN2 inputs (45 total inputs). Best performance was achieved using inputs from an NN1 network that was not identified as the best NN1 network. Best performance was generally provided by networks with relatively few hidden layer PE's (5), and reduced PE connectivity which exploited the spatial relationship of the input lines. NN2 networks yielded significantly lower error and error variation than the NN1 networks that provided their inputs. Like NN1 networks, NN2 networks provided performance superior to that of the heuristic algorithm only for seedlings with morphology similar to that of the training seedlings.

NN3 networks accepted multiple feature inputs (like NN1) from multiple image lines (like NN2). Excellent performance was achieved by many NN3 networks. Thus, fewer statistically significant differences between treatments were observed. Input features were mostly based on connectivity analysis of the seedling image. Clipped feature magnitudes and feature derivatives appeared to be beneficial. The simplest two hidden-layer architecture, with 5 and 3 PE's in the first and second hidden layers, respectively, provided the best performance. Fewer input lines (9 rather than 15 or 21) provided the best performance. A smaller input FOV (15- to 43-mm, rather than 150-mm or greater) provided superior performance. Among the narrow FOV networks, the 29-mm FOV provided the best performance. All NN3 networks used reduced PE connectivity, which exploited the spatial relationship of input lines and reduced computation. NN3 networks yielded performance superior to that of the heuristic algorithm for all Douglas fir seedlings, including those with morphology different from that of the training seedlings. NN3 mean error for Ponderosa pine seedlings was 20 mm, probably due to their significantly different morphology. NN3 error variation, however, was less than that of the heuristic algorithm for these seedlings, as well as the Douglas fir seedlings. NN3 networks more frequently located the root collar within the manually identified stem region than did the heuristic algorithm.

Several NN3 networks yielded comparable high performance. Network w15 performance is reviewed here and compared with that of the heuristic algorithm. This neural network yielded zero mean error and 11-mm error standard deviation for 100 Douglas fir seedlings which had morphology significantly different from that of seedlings used for network training. The network had comparable performance for 50 Douglas fir seedlings similar to those used for training. Heuristic algorithm mean error was -7 mm, and error standard deviation was 12 and 21 mm for the two seedling groups, respectively. The heuristic algorithm located the root collar within the manually identified stem region only 72% of the time, while neural network's "on stem" performance was 93%. The neural network detected the stem region an additional 4% of the time ("found stem" = 97%).

For 150 Ponderosa pine seedlings which had significantly different morphology than that of the Douglas fir training seedlings, the neural network root collar location had a large offset; 20-mm mean error vs. the heuristic algorithm's 3-mm mean error. However, neural network measurement precision was superior to that of the heuristic algorithm; 15-mm vs. 20-mm standard deviation, respectively. Neural network and heuristic algorithm "on stem" performance was 93% and 91%, respectively, for these seedlings. Better results for Ponderosa pine seedlings would be expected if Ponderosa pine seedlings were used as training examples.

Neural networks spanning a range of complexity provided performance superior to that of the heuristic algorithm. Among the architectural variations investigated, configurations with relatively few inputs, easily computed input features, and simpler network architectures generally provided the best performance. These results indicate that a neural network root collar recognition system with a relatively low computational burden can be achieved, consistent with support of real-time seedling inspection. Although the computational burden of the developed networks and feature extraction routines were not quantitatively assessed, it is the author's belief, based on knowledge of processing currently performed by the seedling inspection system, that the methods developed will not add a computational burden prohibitive to real-time assessment. Further, currently available PC's have at least four times more the processing power than the PC used in the existing seedling inspection system.

The research discussed in this dissertation has made several contributions to the body of engineering knowledge. Achievement of the primary objective is the first contribution. Examples of unique neural network applications, given that this is a relatively new computing paradigm, have a synergistic effect as other researchers seek solutions to seemingly un-related problems.

The method used for network training may have broad application. A relatively small percentage of network input data (features) corresponded with the seedling root collar which was to be identified. Many target and object recognition applications similarly seek to identify a signal buried in a complex background.

Reduced PE connectivity architectures used in this work attempted to exploit spatial relationships within seedling images and provided superior performance in terms of both improved root collar location accuracy and reduced computational burden. This approach may be of value in any application where spatial or time-series data are processed. These applications include voice recognition, speech processing, and financial forecasting, to name a few.

Suggested Further Research

Best performance was provided by NN3 type networks. These networks were investigated with only a single output corresponding to the root collar. Since NN1 network results indicate that four-output networks may yield superior performance, investigation of four-output NN3 networks may be worthwhile.

NN1 networks also benefited from two global features; normalized line number, and normalized cumulative area. These and similar features may further improve NN3 performance.

The tendency of wide FOV's to yield lower error variance and narrow FOV's to yield lower mean error might be exploited. Coarse, wide-FOV processing might initially be used to locate the stem region. Narrow-FOV processing of every image line within the stem region could then be used to locate the root collar.

A simple post-processing algorithm was used to integrate the network output from each image line and identify the root collar location. Post-processing variations should be investigated. Should maximum network response be used as the root collar location?

Several of the developed networks provided performance superior to that of a heuristic algorithm used in the seedling inspection system. Integration of NN3 type networks into the seedling inspection system should therefore be of value.

REFERENCES

Anderson, J.A. 1988. General introduction. Neurocomputing: Foundations of research. Eds. J.A. Anderson and E. Rosenfeld. MIT Press, Cambridge, MA.

Baum, E.B. and D. Haussler. 1989. What size net gives valid generalization? Neural Computation, 1(1):151-160.

Burks, T.F., S.A. Shearer, and R.S. Gates. 1994. Neural network classification of plant canopy images from texture features. ASAE Paper No. 943510, ASAE, St. Joseph, MI.

Carpenter, G.A. 1989. Neural network models for pattern recognition and associative memory. Neural Networks, 2(4):243-257.

Casasent, D. and S. Natarajan. 1995. A classifier neural net with complex-valued weights and square-law nonlinearities. Neural Networks, 8(6):989-998.

Casasent, D. and M.A. Sipe. 1997. Neural net classification of X-ray pistachio nut data. In: Optics in Agriculture, Forestry, and Biological Processing II, George E Meyer, James A. DeShazer, eds., Proc. SPIE 2907, pp 217-227.

Data Translation. 1990. DT2856 User Manual. Data Translation, Inc., Marlboro, MA.

Davis, D.B. and J. Scholtes. 1995. Machine vision development and use in seedling quality monitoring inspection. In: Landis, T.D., Cregg, B., tech. coords. National Proceedings, Forest and Conservation Nursery Associations. General Technical Report PNW-GTR-365, USDA Forest Service, Pacific Northwest Research Station, Portland, OR, pp 75-79.

Duddles, R.E. and P.W. Owston. 1990. Performance of conifer stocktypes on national forests in the Oregon and Washington coast ranges. In: Target Seedling Symposium: Proceedings, Combined Meeting of the Western Forest Nursery Associations. General Technical Report RM-200, USDA Forest Service, Fort Collins, CO, pp 263-268.

Forward, P.W. 1982. Stock production specifications - bare root stock. In: Artificial Regeneration of Conifers in the Upper Great Lakes Region. Michigan Technological University, Houghton, MI, pp 260-268.

Ghazanfari, A. and J. Irudayaraj. 1994. Multi-structure neural networks (MSNN) for classification of pistachio nuts. ASAE Paper No. 946579, ASAE, St. Joseph, MI.

Golomb, B. and T. Sejnowski. 1995. Sex recognition from faces using neural networks. In: Applications of Neural Networks, A. Murray, ed. Klewer Academic Publishers, Boston, MA, pp 71-92.

Gonzalez, C.G. and R.E. Woods. 1992. Digital Image Processing. Addison-Wesley Publishing Co., Reading, MA.

Jain, A.K. 1989. Fundamentals of Digital Image Processing. Prentice Hall, Englewood Cliffs, NJ.

Kranzler, G.A. and M.P. Rigney. 1989. Feasibility of machine vision for tree seedling grading and root growth measurement. Report to: U.S.D.A. Forest Service, Missoula Equipment Development Center. Missoula, MT.

Kranzler, G.A. and M.P. Rigney. 1994. Seedling inspection equipment development I. Report to: Oregon Department of Forestry, D.L. Phipps State Forest Nursery, Elkton, OR.

Kranzler, G.A. and M.P. Rigney. 1996. Seedling inspection equipment development II. Report to: Oregon Department of Forestry, D.L. Phipps State Forest Nursery, Elkton, OR.

Mangold, D.R., Moulton, R.J. and Snellgrove, J.D. 1992. Tree planting in the United States - 1991. Cooperative Forestry, State and Private Forestry, USDA Forest Service, 14 pp.

May, J.T. 1985. Seedling quality, grading, culling and counting. In: Lantz, C.W. (Editor and Compiler), Southern Pine Nursery Handbook. USDA Forest Service, Southern Region, pp 9-1 - 9-10.

Menes, P.A. and G.H. Mohammed. 1995. Identifying the root collar on forest tree seedlings. The Forestry Chronicle, 71(3):304-310.

Mexal, J.G. and T.D. Landis. 1990. Chapter 3: Target Seedling Concepts: Height and Diameter. In: Target Seedling Symposium: Proceedings, Combined Meeting of the Western Forest Nursery Associations. General Technical Report RM-200, USDA Forest Service, Fort Collins, CO.

Mowry, D. 1995. Current developments in the prevention and treatment of repetitive motion injuries of the upper extremity. In: Landis, T.D., Cregg, B., tech. coords. National Proceedings, Forest and Conservation Nursery Associations. General Technical Report PNW-GTR-365. USDA Forest Service, Pacific Northwest Research Station, Portland, OR, pp 8-12.

NeuralWare. 1995a. Neural Computing: A Technology Handbook for Professional II/Plus and NeuralWorks Explorer. NeuralWare, Inc., Pittsburgh, PA.

NeuralWare. 1995b. DataSculptor: Data Transformation and Analysis Tool for Windows. NeuralWare, Inc., Pittsburgh, PA.

Okholm, D.J. and R.D. Abriel. 1994. Directory of Forest and Conservation Tree Nurseries in the United States. U.S.D.A. Forest Service. R6-CP-TP-02-94.

Pao, Y. 1989. Adaptive Pattern Recognition and Neural Networks. Addison-Wesley Publishing Co., Inc. New York.

Park, B., Y.R. Chen, A.D. Whittaker, R.K. Miller, and D.S. Hale. 1994. Neural network modeling for beef sensory evaluation. Trans. of ASAE, 37(5):1547-1553.

Raudys, S.J. and A.K. Jain. 1991. Small sample size effects in statistical pattern recognition: recommendations for practitioners. IEEE Trans. on Pattern Analysis and Machine Intelligence, 13(3):252-264.

Ritchie, G.A. 1984. Assessing Seedling Quality. In: Forest Nursery Manual: Production of Bareroot Seedlings. Martinus Nijhoff/Dr. W. Junk Publishers, Boston, pp 243-259.

Rigney, M.P. and G.A. Kranzler. 1988. Machine vision for grading southern pine seedlings. Trans. of ASAE, 31(2):642-646.

Rigney, M. P. and G.A. Kranzler. 1989. Performance of a machine vision based tree seedling grader. ASAE Paper No. 89-3007, ASAE, St. Joseph, MI.

Rigney, M P. and G.A. Kranzler. 1993. Line-scan inspection of conifer seedlings. In: Optics in Agriculture and Forestry, James A. DeShazer, George E Meyer, eds., Proc. SPIE 1836, pp 166-174.

Rigney, M.P. and G.A. Kranzler. 1995. Machine vision system for measuring conifer seedling morphology. In: Optics in Agriculture, Forestry, and Biological Processing, George E Meyer, James A. DeShazer, eds., Proc. SPIE 2345, pp 26-35.

Romaniuk, S.G. and L.W. Hall. 1993. Divide and conquer neural networks. Neural Networks, 6(8):1105-1116.

Rose, R. and D.L. Haase. 1995. The target seedling concept: Implementing a program. In: Landis, T.D.; Cregg, B., tech. coords. National Proceedings, Forest and Conservation Nursery Associations. General Technical Report PNW-GTR-365. USDA Forest Service, Pacific Northwest Research Station, Portland, OR, pp 75-79.

Rumelhart, D.E., G.E. Hinton, and J.L. McClelland. 1986a. A general framework for parallel distributed processing. In: Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Volume 1, D.E. Rumelhart and J.L. McClelland, eds. MIT Press, Cambridge, MA.

Rumelhart, D.E., G.E. Hinton, and J.L. McClelland. 1986b. Learning internal representations by error propagation. In: Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Volume 1, D.E. Rumelhart and J.L. McClelland, eds. MIT Press, Cambridge, MA.

Scholtes, J.R. 1994. Machine vision development: Its use at a forest seedling nursery. Combined Proceedings International Plant Propagators' Society, Vol. 44.

Suh, Sang-Ryong and G.E. Miles. 1988. Measurement of morphological properties of tree seedlings using machine vision and image processing. ASAE Paper No. 88-1542, ASAE, St. Joseph, MI.

Vincent, J.M. 1995. Face finding in images. In: Applications of Neural Networks, A. Murray, ed. Klewer Academic Publishers, Boston, pp. 35-70.

Wallersteiner, U. 1988. Cumulative trauma disorders in forest nursery workers. In: Proceedings, Combined Meeting of the Western Forest Nursery Associations. General Technical Report RM-167, USDA Forest Service, Fort Collins, CO, pp 75-76.

Wang, Deliang. 1993. Pattern recognition: Neural networks in perspective. IEEE Expert, 8(4):52-60.

Weymaere, N. and J. Martens. 1991. A fast and robust learning algorithm for feedforward neural networks. Neural Networks, 4(3):361-369.

Wilhoit, J.H., L.J. Kutz, D.E. Fly, and D.B. South. 1994. PC-based multiple camera machine vision systems for pine seedling measurements. Applied Engineering in Agriculture, 10(6):841-847.

APPENDIXES

APPENDIX A--FEATURE COMPUTATION FORMULAS

APPENDIX B--FEATURE ILLUSTRATIONS FOR FIVE SEEDLINGS

APPENDIX C--RESPONSE OF SELECTED NEURAL NETWORKS FOR FIVE SEEDLINGS

APPENDIX D--SUMMARY OF NEURAL NETWORK FEATURES, ARCHITECTURE, AND PERFORMANCE

APPENDIX E--STATISTICAL ANALYSIS OF NEURAL NETWORK PERFORMANCE

APPENDIX A--FEATURE COMPUTATION FORMULAS

Feature names, definitions, and computational formulas are presented in this appendix and summarized in Tables A1 and A2. Values of selected features, illustrating their relationship to the seedling stem region and root collar location, are plotted for five seedlings (Fig. 4) in Appendix B, Figures B1 through B19.

General Functions Applied to Feature Computation.

Many features were extracted from a local window (*w*) encompassing 5 to 21 image lines. A few features were extracted from a similar window with broader scope, 161 image lines. Window size (*w*) is an odd number. Results are associated with the line at the window center (evaluation line).

Average: average value of a line feature over a set of adjacent lines, {-(*w*-1)/2,...,0,...,+(*w*-1)/2} centered about the evaluation line, *l*. This feature was computed to reduce "noise" and to broaden the scope of a local feature by integrating feature data within the neighborhood of the evaluation line.

$$avg(feature, w) = \frac{\sum_{i=l-\frac{(w-1)}{2}}^{i=l+\frac{(w-1)}{2}} feature_i}{w}$$

<u>Minimum</u>: minimum value of a line feature over a set of adjacent lines, *w*, centered about the evaluation line, *l*.

$$min(feature, w) = min(feature_{l-\frac{(w-1)}{2}}, \ldots, feature_{l+\frac{(w-1)}{2}})$$

<u>Maximum</u>: maximum value of a line feature over a set of adjacent lines, *w*, centered about the evaluation line, *l*.

$$max(feature, w) = max(feature_{l-\frac{(w-1)}{2}}, \ldots, feature_{l+\frac{(w-1)}{2}})$$

<u>Delta</u>: difference operator computing the slope of a line feature based on the feature values at the limits of a window, *w*, centered about the evaluation line, *l*. The derivative of many features was observed to be close to zero near the root collar (RC). Further, the derivative value is often negative immediately above the RC and positive below the RC.

$$delta(feature, w) = feature_{l+\frac{(w-1)}{2}} - feature_{l-\frac{(w-1)}{2}}$$

<u>Difference</u>: difference operator applied to two features extracted from or evaluated at the same line.

$$diff(feature_1, feature_2) = feature_1 - feature_2$$

<u>Clip</u>: limit the magnitude of a feature value by replacing large values with a specified threshold.

$$clip(threshold, feature) = if\,(feature < threshold)\; feature; \quad else\; threshold$$

General Description of Features Computed for Neural Network Input.

Normalized Line Number (Nline): line number divided by the total number of lines for each seedling.

Runs: the number of runs on a line (Fig B1).

Area: the sum of run lengths on a line (Fig. B5).

Normalized Cumulative Area (NCArea): cumulative area from the top of the seedling to the current line divided by the total seedling area (Fig. B7).

Delta Normalized Cumulative Area (DelNCArea): slope of the normalized cumulative area curve (Fig. B8).

Span: distance from beginning of the first run to end of the last run on a line (Fig. B9).

Inverted Runs (InvRuns), Inverted Area (InvArea), Inverted Span (InvSpan): one divided by Runs (Fig. B4), Area (Fig. B6), and Span, respectively.

Smooth Runs (SmRuns), Smooth Area (SmArea), Smooth Span (SmSpan): average of Runs (Fig. B2), Area, and Span, respectively, within an 11-line window.

Delta Smooth Runs (DelSmRuns), Delta Smooth Area (DelSmArea), Delta Smooth Span (DelSmSpan): slope of SmRuns (Fig. B3), SmArea, and SmSpan, respectively.

Connected Run: a run which is "connected" to the previous line, *l*-1, as determined by connectivity analysis. A connected run shares a column index with a connected run on the previous line. Only the three longest connected runs are retained. Connectivity analysis is initialized with the longest run from the first image line.

Connected Runs (ConRun): number of connected runs on a line (Fig. B13).

Connected Area (ConArea): sum of run lengths of connected runs on a line.

Delta Connected Area (DelConC100): slope of ConArea clipped to a magnitude of 100.

Un-Connected Area (UnCon): difference between Area and ConArea on a line (Fig. B14).

Offset Connected Area (ConMn41, ConMn61, ConMn41C50): difference between ConArea on a line and the minimum ConArea within a window (41 or 61 lines), (Fig. B15).

Offset Smooth Connected Area (ConA11): difference between average ConArea within a local window (11 lines) and the minimum ConArea within a broad window (161 lines).

Clipped Offset Smooth Connected Area (ConA11C50, ..., ConA21C25): difference between average ConArea within a local window (5-21 lines) and the minimum ConArea within a broad window (161 lines), clipped to a magnitude of 25 or 50 (Fig. B16).

Clipped Minimum Connected Area (ConMn11C50, ..., ConMn21C25): difference between the minimum ConArea within a local window (5-21 lines) and the minimum ConArea within a broad window (161 lines), clipped to a magnitude of 25 or 50.

Close Run: a run which is not "connected" but has an endpoint within 5 mm (50 pixels) of a connected run on the same line.

Close Area (CloseArea): sum of run lengths of close runs (Fig. B17).

Delta Close Area (DelCloseC100): slope of CloseArea clipped to a magnitude of 100.

Smooth Close Area (CloseA11): the average CloseArea within an 11 line window.

Clipped Smooth Close Area (CloseA11C50, ..., CloseA21C25): the average CloseArea within a window (5-21 lines) clipped to a magnitude of 25 or 50 (Fig. B18).

Clipped Maximum Close Area (CloseMx11C50, ..., CloseMx21C25): the maximum CloseArea within a window (5-21 lines) clipped to a magnitude of 25 or 50 (Fig. B19).

Log Second Central Moment (M2): the natural logarithm of the second order central moment computed from all runs on a line (Fig. B10).

Log Third Central Moment (M3): the natural logarithm of the absolute value of the third central moment computed from all runs on a line (Fig. B11).

Run-Length Histogram (Hist1, ..., Hist8): the number of runs on a line having a length within the limits specified for the histogram bin (Fig. B12).

Table A1. Feature summary

Feature	Figure	Networks	Definition or Formula
Nline		L4,L5,L6,L7	line / lastline
NCArea	B7	L4,L5	sum(Area, 1, line) / sum(Area, 1, lastline)
DelNCArea	B8	L5	delta(NCArea, 11)
Runs	B1	L1,L2,L4,L5	number of seedling runs
Area	B5	-	sum of seedling run lengths
Span	B9	-	lastrun.end - firstrun.start
InvRuns	B4	L1,L2,L4,L5	1/Runs
InvArea	B6	L1,L2,L4,L5	1/Area
InvSpan		L1,L2,L4,L5	1/Span
SmRuns	B2	L1,L4,L5,L7	avg(Runs, 11)
SmArea		L1,L4-5,L7,M1-6	avg(Area, 11)
SmSpan		L1,L4,L5,L7	avg(Span, 11)
DelSmRuns	B3	L1,L4,L5	delta(SmRuns, 11)
DelSmArea		L1,L4,L5	delta(SmArea, 11)
DelSmSpan		L1,L4,L5	delta(SmSpan, 11)
M2	B10	L2,L5	ln(2nd central moment)
M3	B11	L2,L5	ln(abs(3rd central moment))
Hist1	B12	L3,L6	number of runs with length less than 1 mm
Hist2	B12	L3,L6	number of runs, 1-2 mm in length
Hist3	B12	L3,L6	number of runs, 2-4.5 mm in length
Hist4		L3,L6	number of runs, 4.5-7 mm in length
Hist5		L3,L6	number of runs, 7-9.5 mm in length
Hist6	B12	L3,L6	number of runs, 9.5-12 mm in length
Hist7	B12	L3,L6	number of runs, 12-50 mm in length
Hist8		L3,L6	number of runs, greater than 50 mm in length

Table A2. Summary of connectivity analysis features

Feature	Figure	Networks	Definition or Formula
ConRuns	B13	L7	number of connected runs, max=3
ConArea		L7,M1,M2,M3	sum of connected run lengths
DelConC100		W16,W17	clip(100, delta(ConArea,11))
UnCon	B14	L7	diff(Area, ConArea)
ConMn41	B15	M4,M5	diff(ConArea, min(ConArea,41))
ConMn61		M6	diff(ConArea, min(ConArea,61))
ConMn41C50		M7,M8	clip(50, ConMn41)
ConA11		W1	diff(avg(ConArea,11), min(ConArea,161))
ConA11C50	B16	W2	clip(50, diff(avg(ConArea,11), min(ConArea,161)))
ConA11C25		W5,W14,W16	clip(25, diff(avg(ConArea,11), min(ConArea,161)))
ConA5C25		W7,W12	clip(25, diff(avg(ConArea,5), min(ConArea,161)))
ConA15C25		W8	clip(25, diff(avg(ConArea,15), min(ConArea,161)))
ConA21C25		W10	clip(25, diff(avg(ConArea,21), min(ConArea,161)))
ConMn11C50		W3	clip(50, diff(min(ConArea,11), min(ConArea,161)))
ConMn11C25		W4,W15,W17	clip(25, diff(min(ConArea,11), min(ConArea,161)))
ConMn5C25		W6	clip(25, diff(min(ConArea,5), min(ConArea,161)))
ConMn15C25		W9	clip(25, diff(min(ConArea,15), min(ConArea,161)))
ConMn21C25		W11	clip(25, diff(min(ConArea,21), min(ConArea,161)))
CloseArea	B17	L7,M1-M7	sum close connected runs
DelCloseC100		W16,W17	clip(100, delta(CloseArea,11))
CloseAreaC50		M8	clip(50, CloseArea)
CloseA11		W1	avg(CloseArea, 11)
CloseA11C50	B18	W2	clip(50, avg(CloseArea,11))
CloseA11C25		W5,W14,W16	clip(25, avg(CloseArea,11))
CloseA5C25		W7,W12	clip(25, avg(CloseArea,5))
CloseA15C25		W8	clip(25, avg(CloseArea,15))
CloseA21C25		W10	clip(25, avg(CloseArea,21))
CloseMx11C50	B19	W3	clip(50, max(CloseArea,11))
CloseMx11C25		W4,W15,W17	clip(25, max(CloseArea,11))
CloseMx5C25		W6	clip(25, max(CloseArea,5))
CloseMx15C25		W9	clip(25, max(CloseArea,15))
CloseMx21C25		W11	clip(25, max(CloseArea,21))

APPENDIX B--FEATURE ILLUSTRATIONS FOR FIVE SEEDLINGS

Figure B1. Feature "runs" for 5 seedlings.

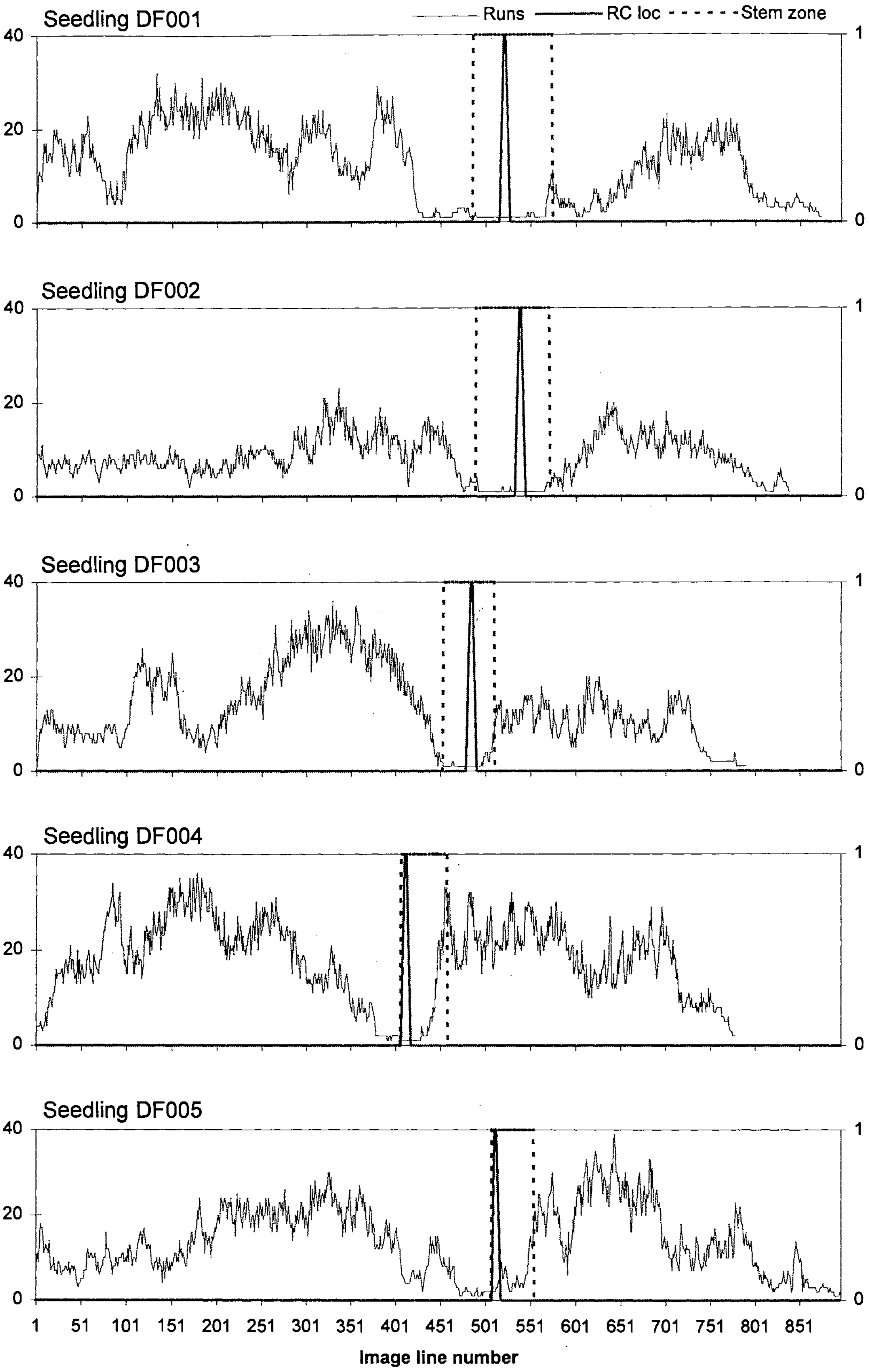


Figure B2. Feature "smooth runs" for 5 seedlings.

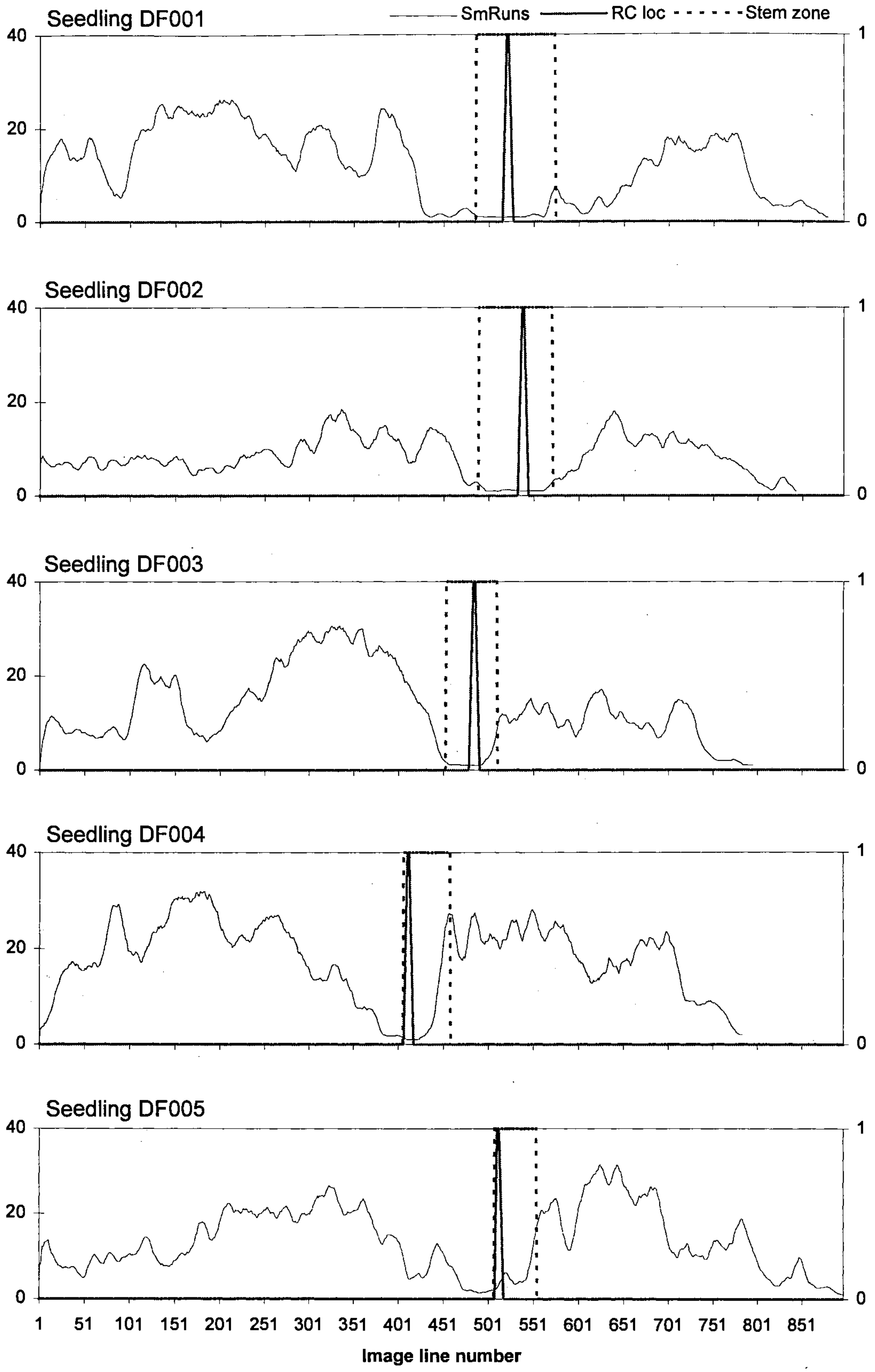


Figure B3. Feature "delta smooth runs" for 5 seedlings.

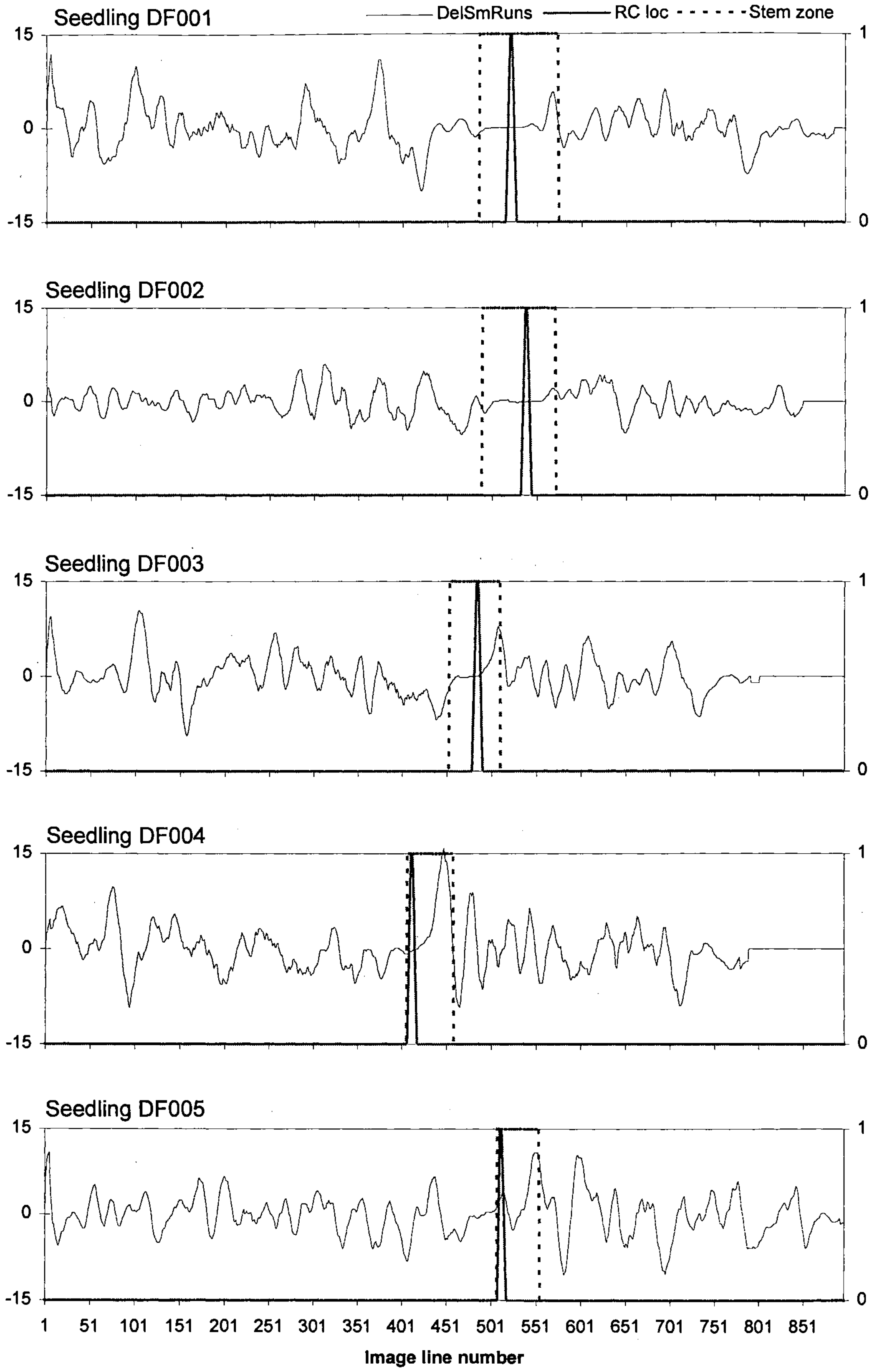


Figure B4. Feature "inverted runs" for 5 seedlings.

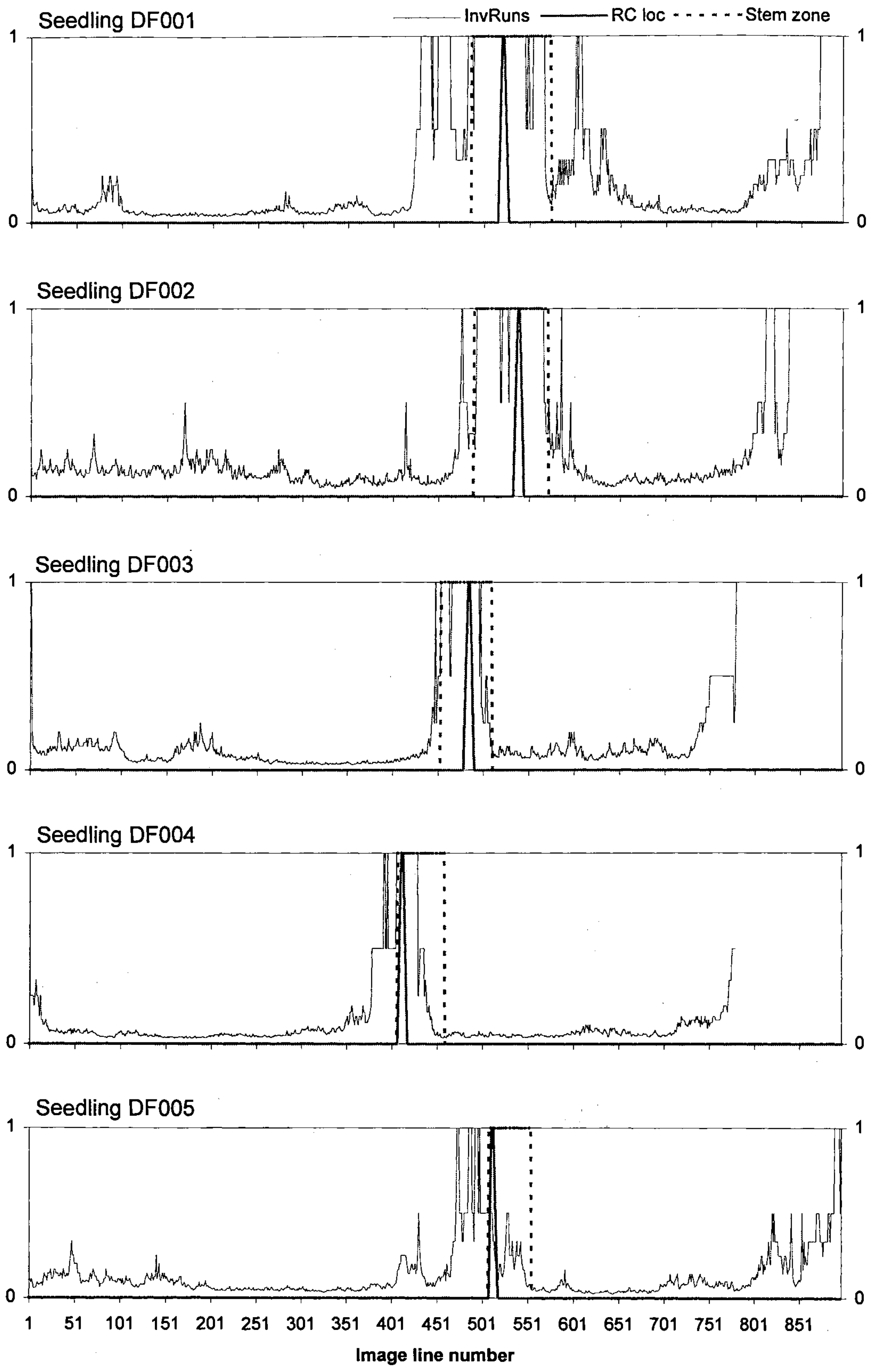


Figure B5. Feature "area" for 5 seedlings.

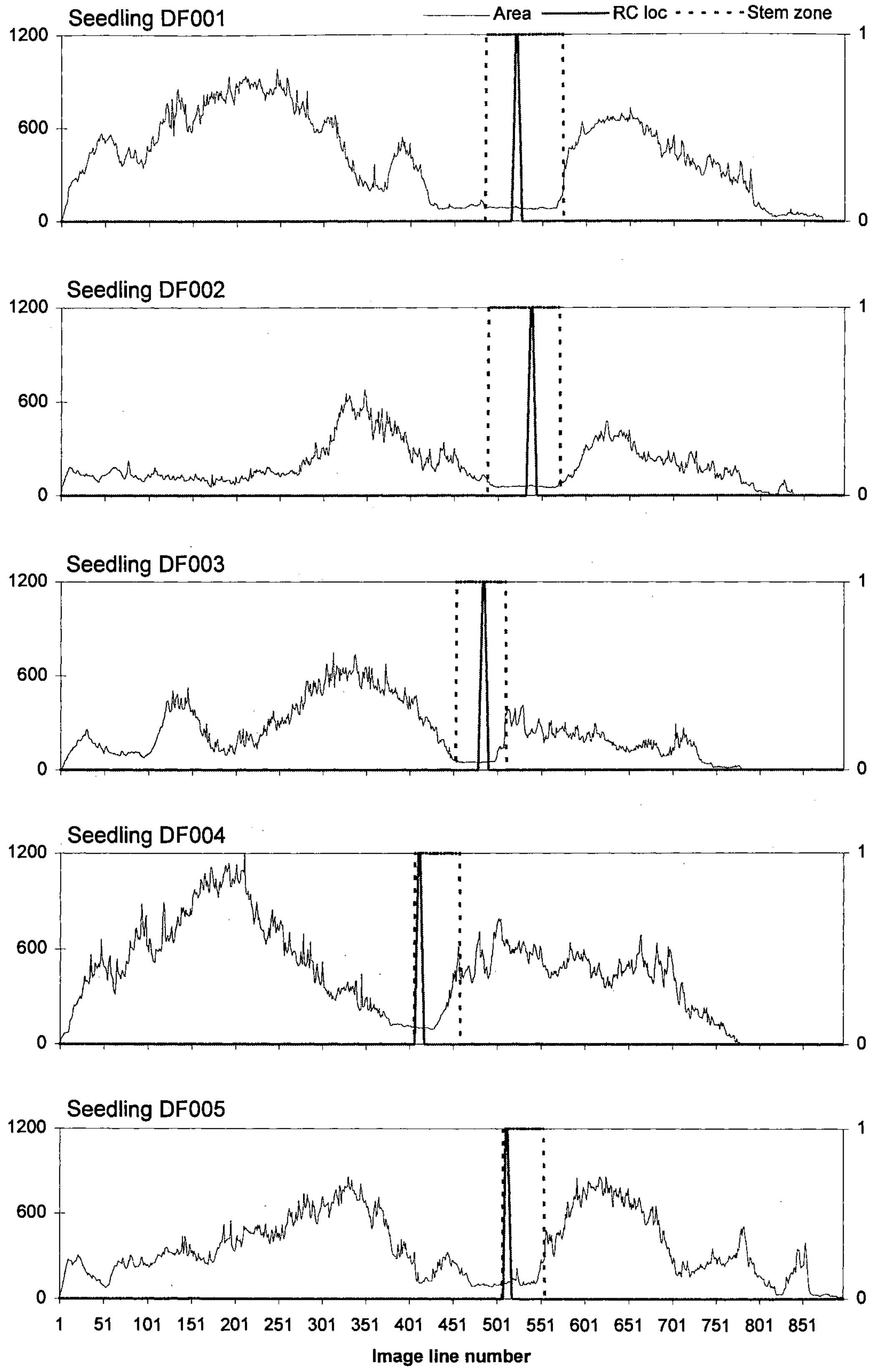


Figure B6. Feature "inverted area" for 5 seedlings.

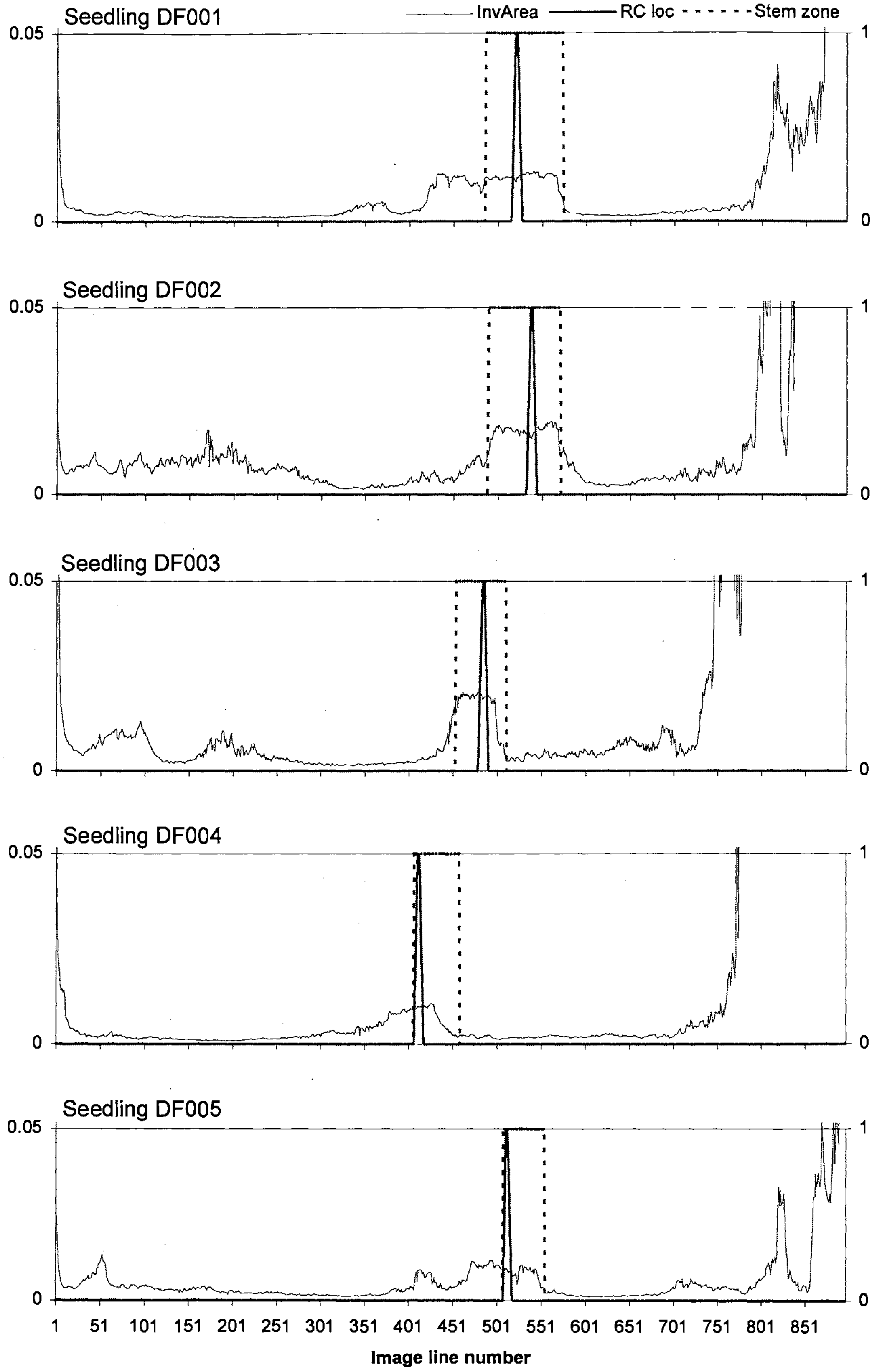


Figure B7. Feature "normalized cumulative area" for 5 seedlings.

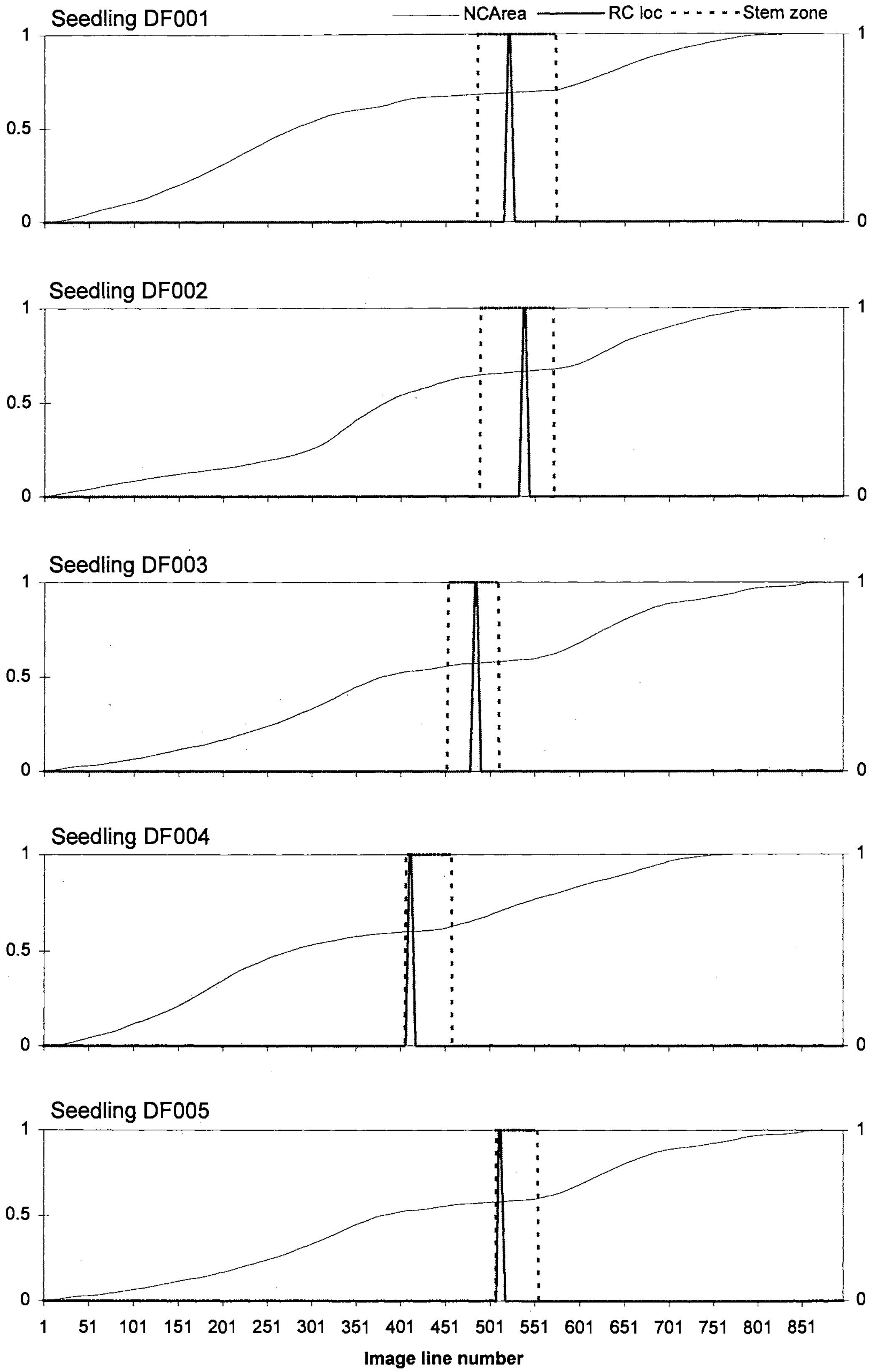


Figure B8. Feature "delta normalized cumulative area" for 5 seedlings.

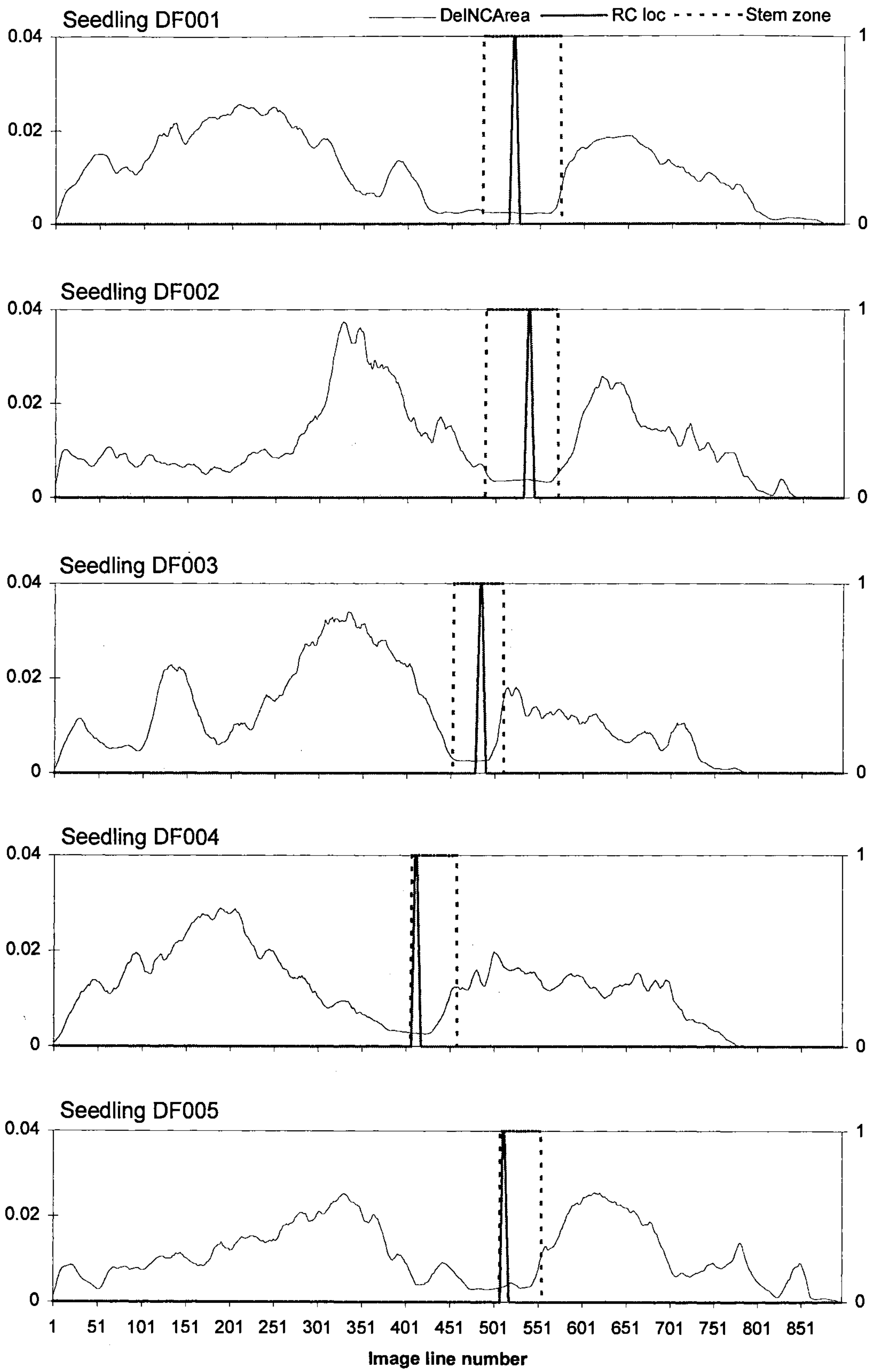


Figure B9. Feature "span" for 5 seedlings.

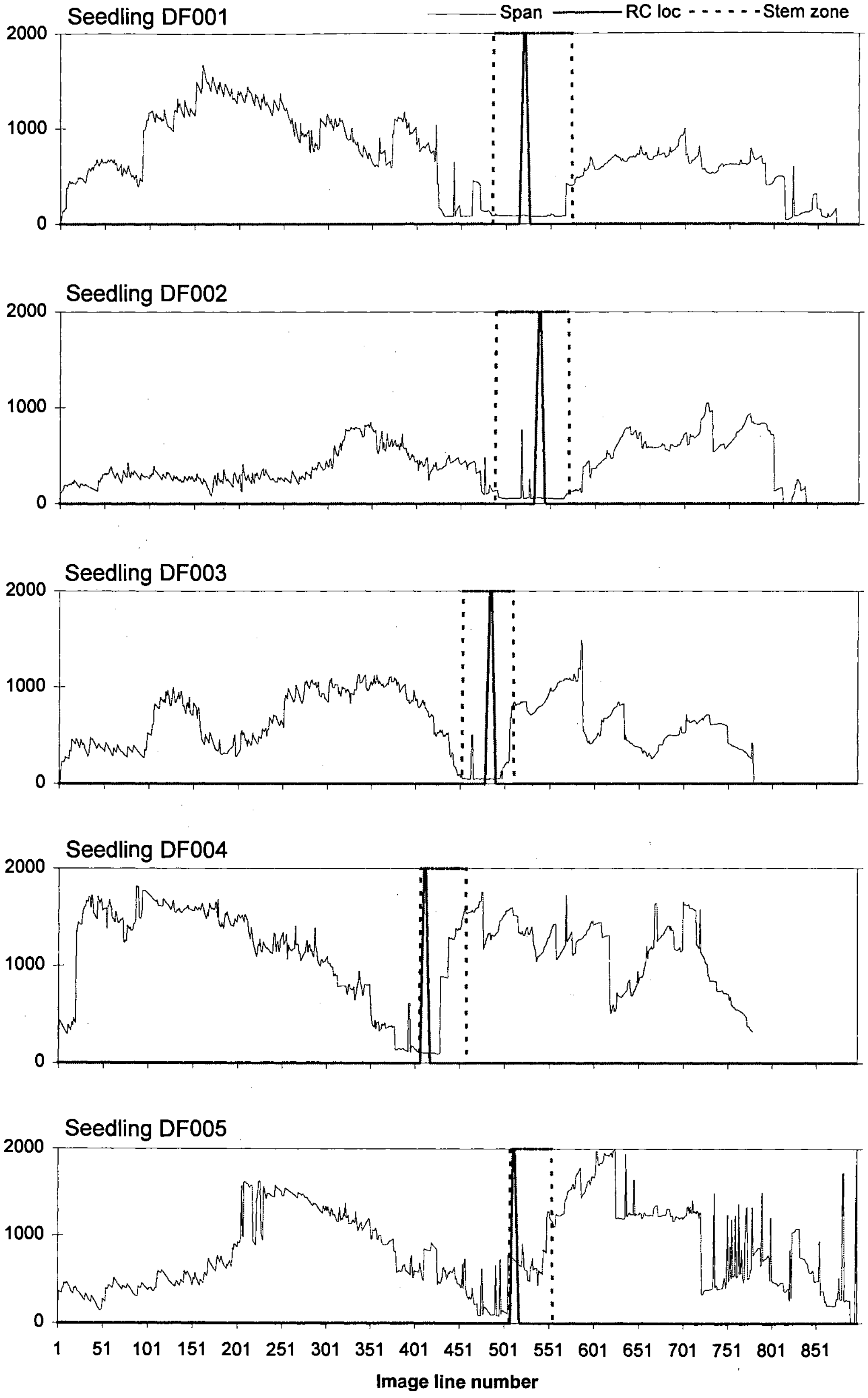


Figure B10. Feature "log 2nd central moment" for 5 seedlings.

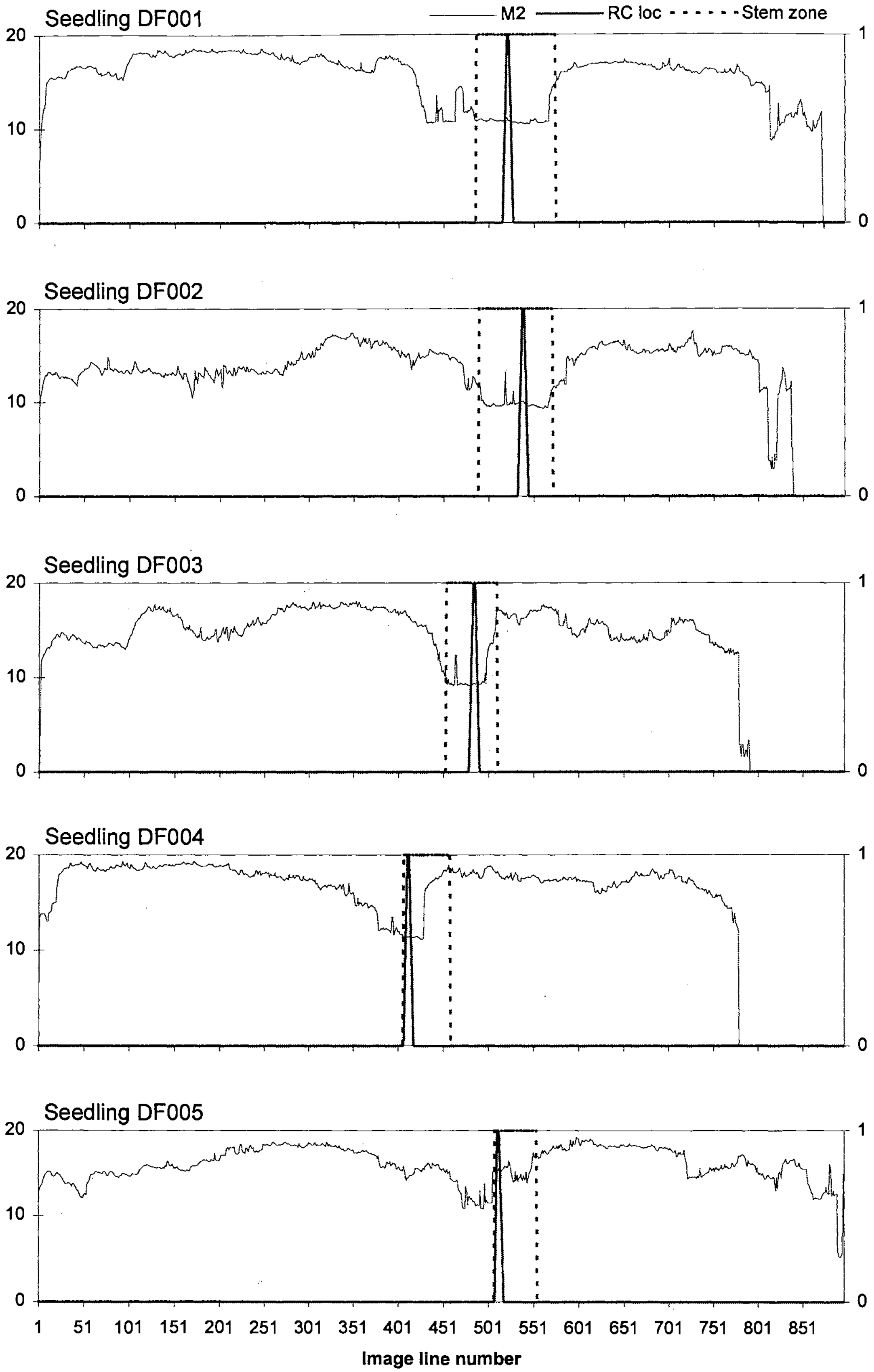


Figure B11. Feature "log absolute 3rd central moment" for 5 seedlings.

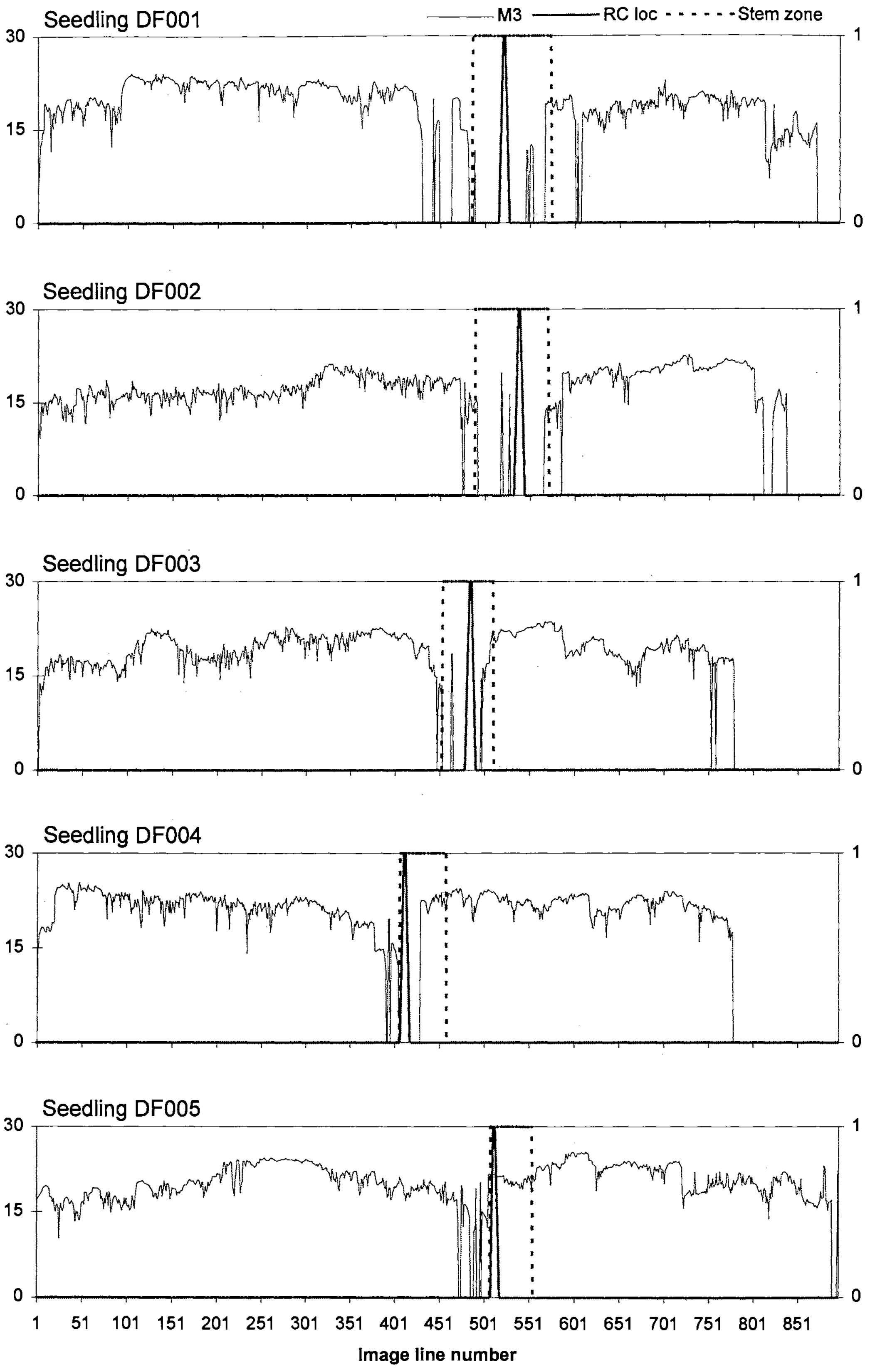


Figure B12. Histogram features for Seedling DF001.

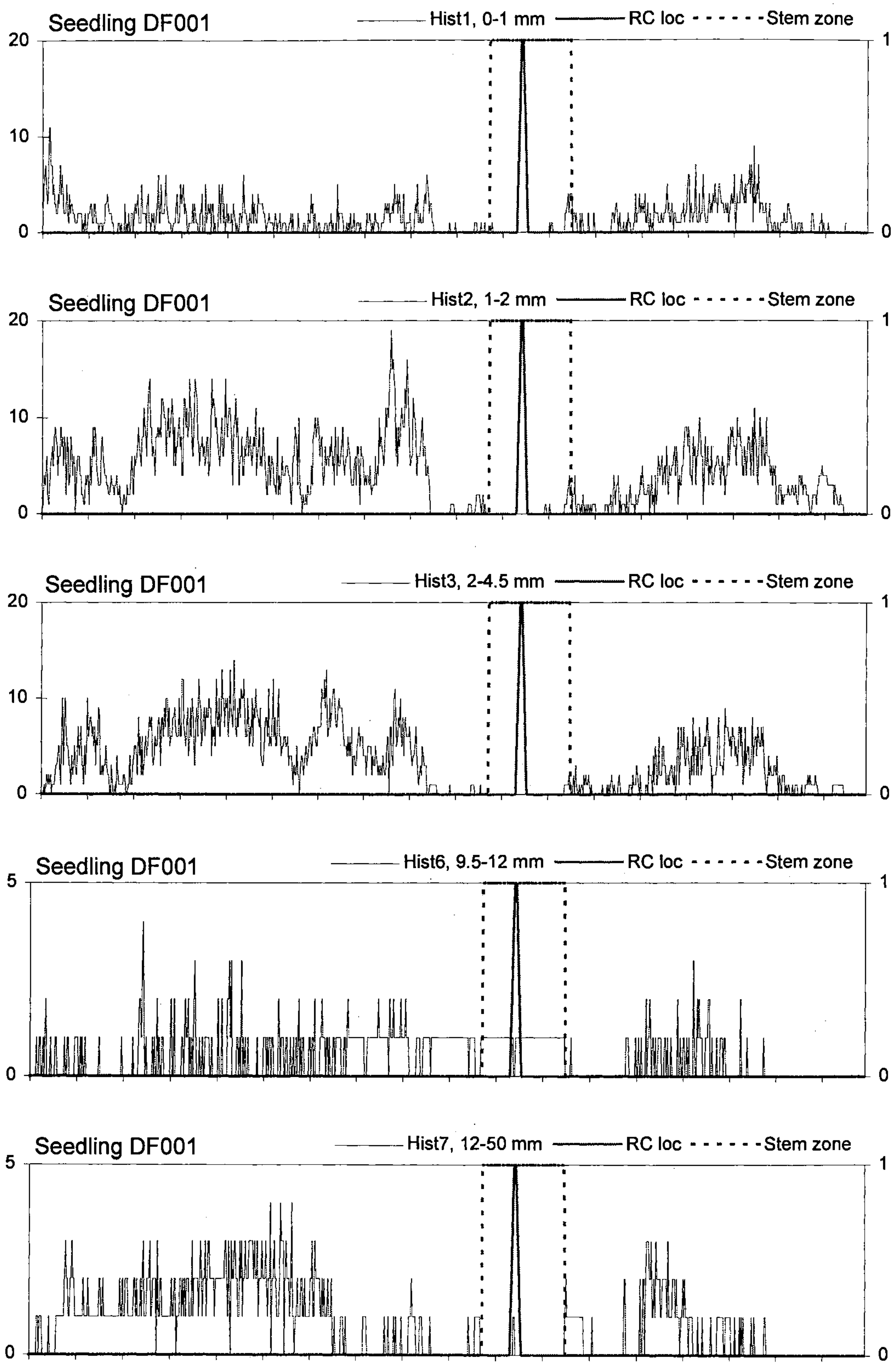


Figure B13. Feature "connected runs" for 5 seedlings.

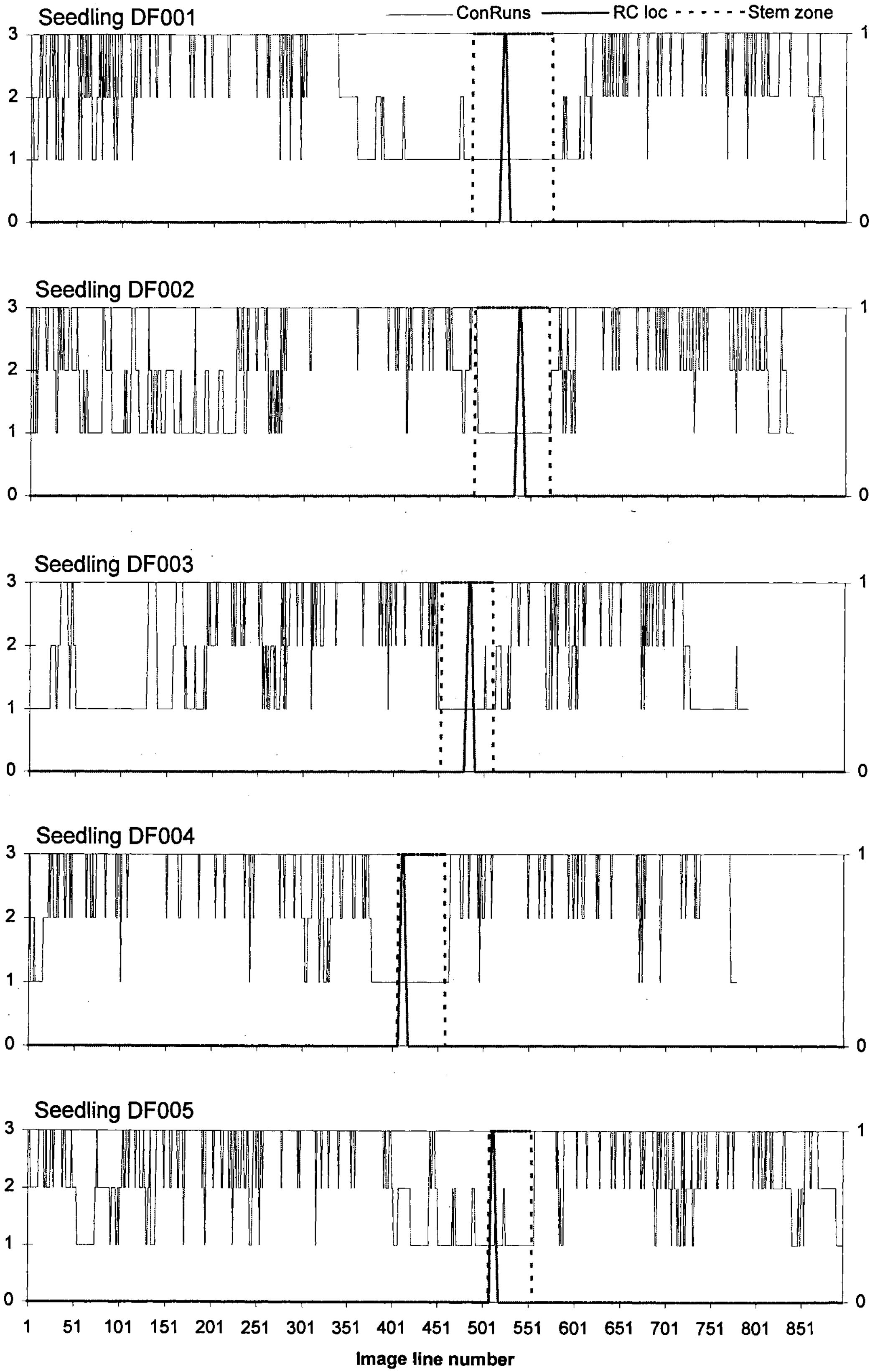


Figure B14. Feature "un-connected area" for 5 seedlings.

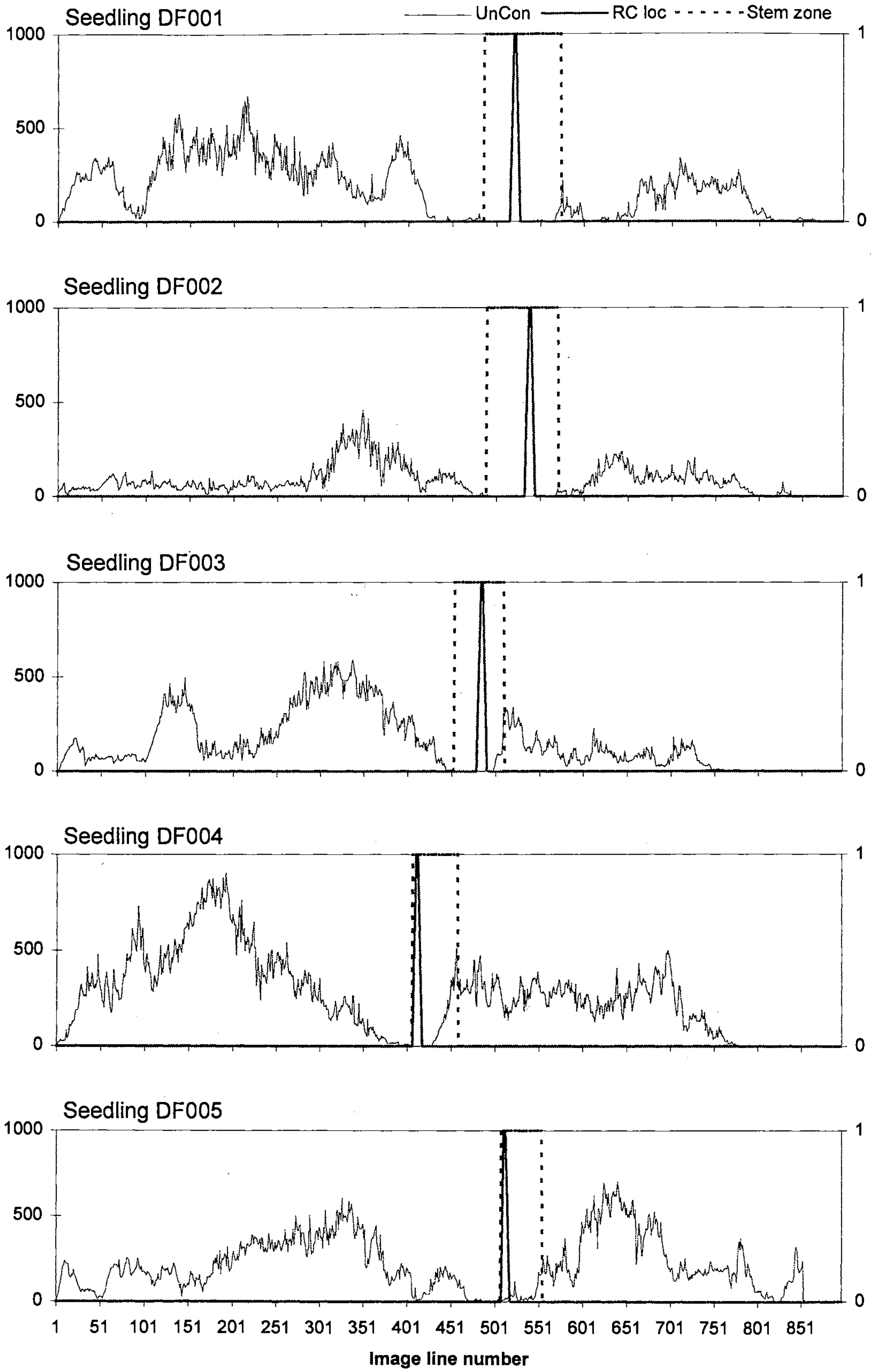


Figure B15. Feature "offset connected area" for 5 seedlings.

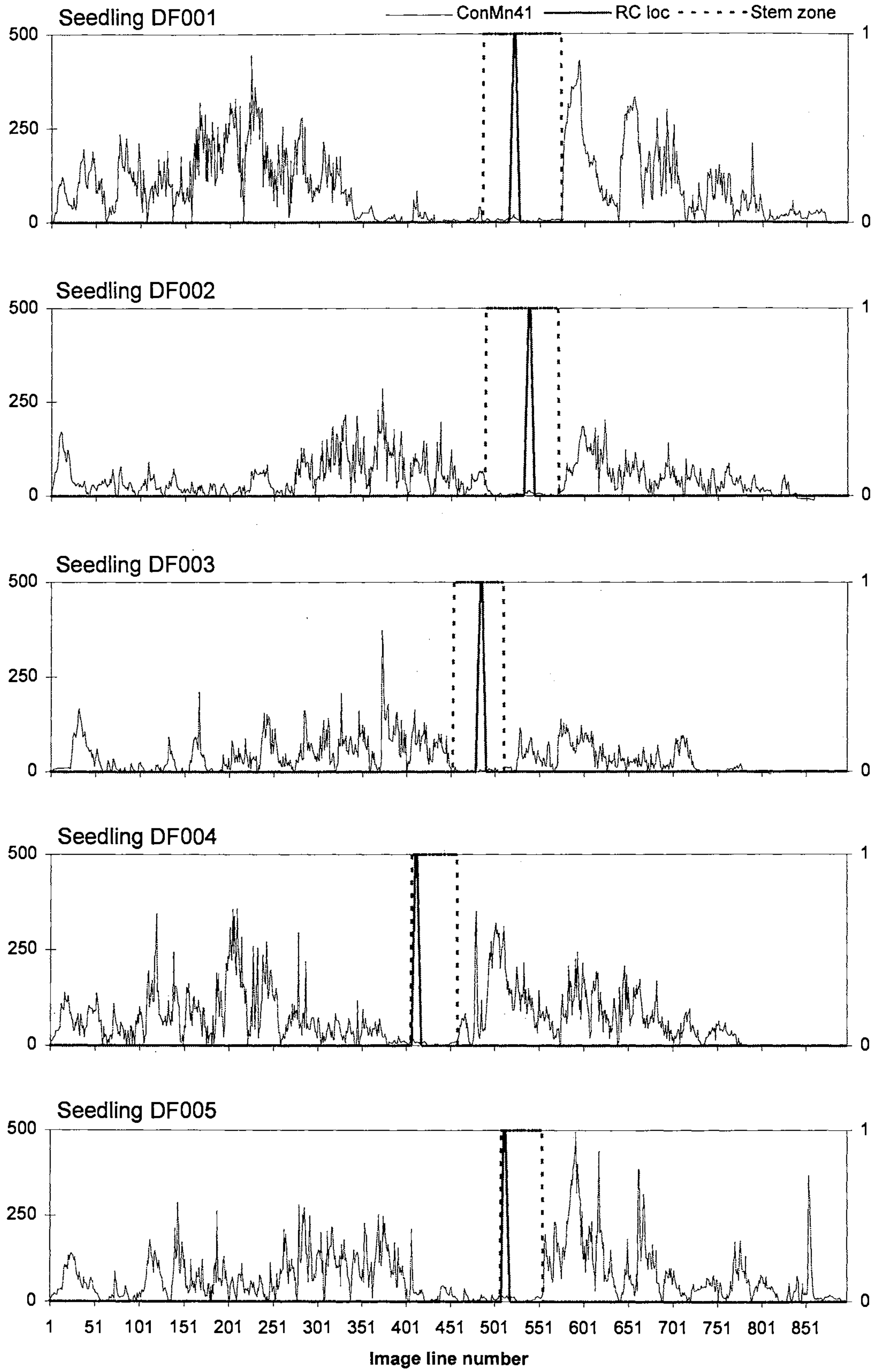


Figure B16. Feature "clipped offset smooth ConArea" for 5 seedlings.

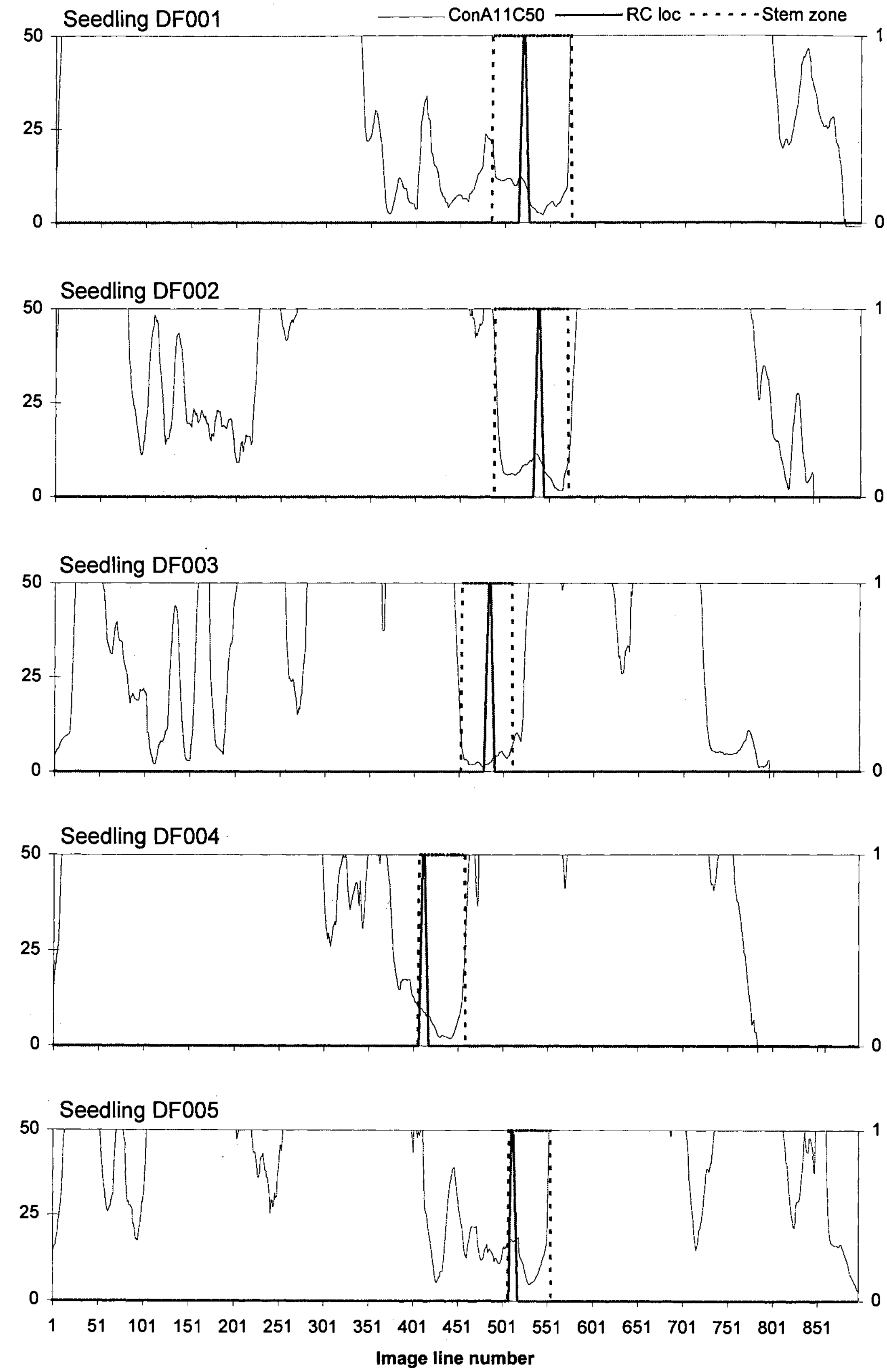


Figure B17. Feature "close connected area" for 5 seedlings.

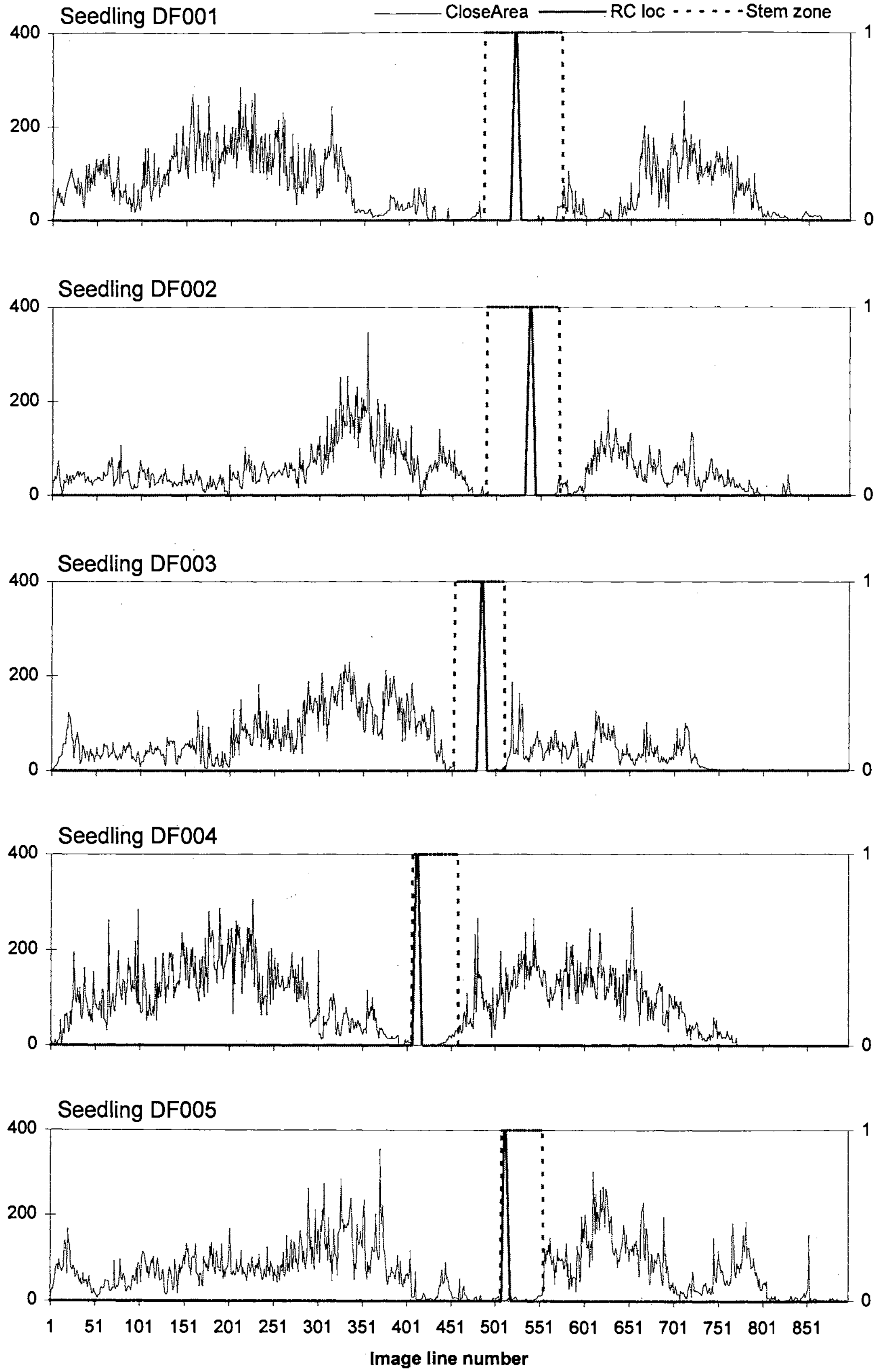


Figure B18. Feature "clipped smooth close area" for 5 seedlings.

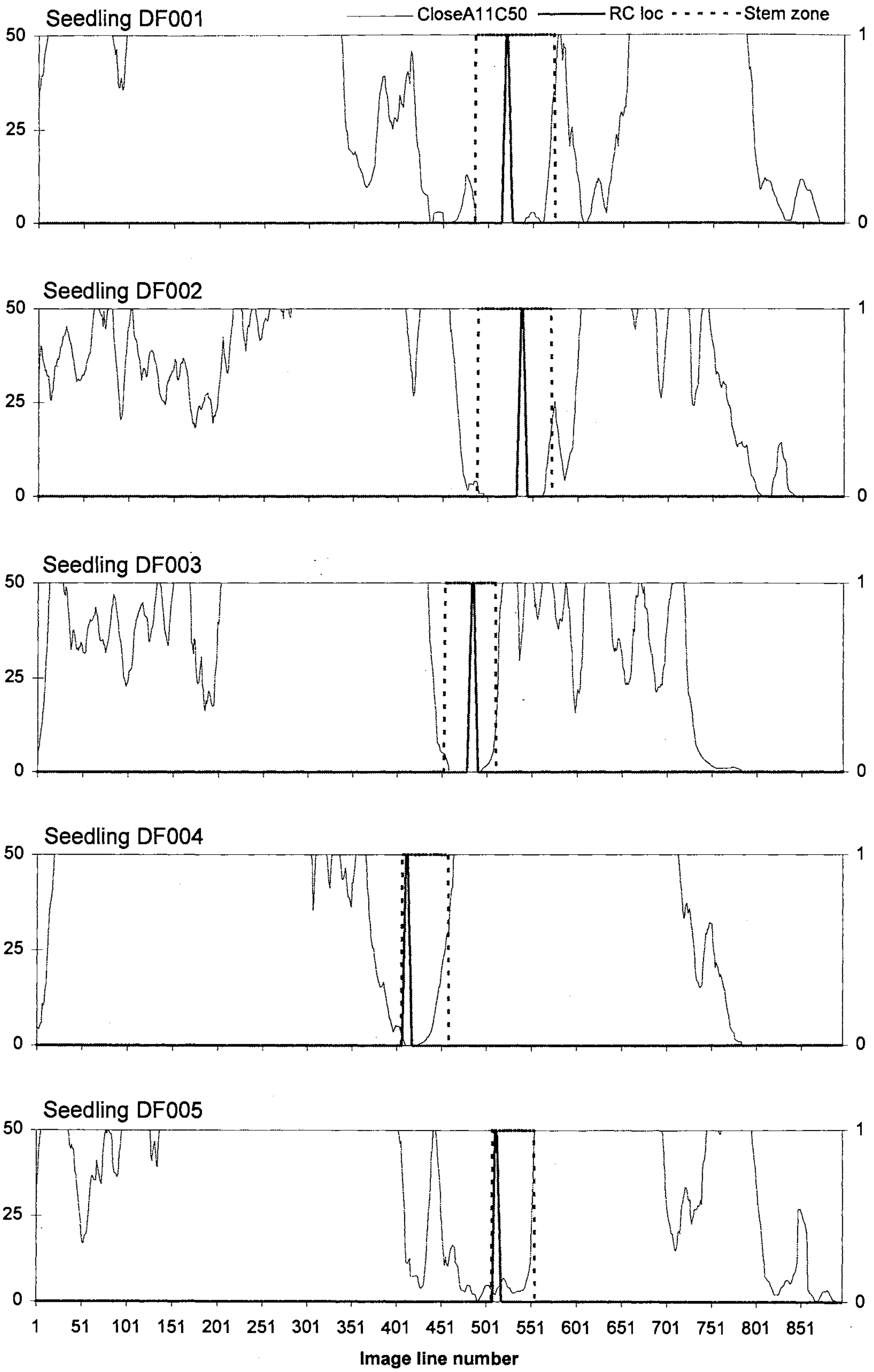
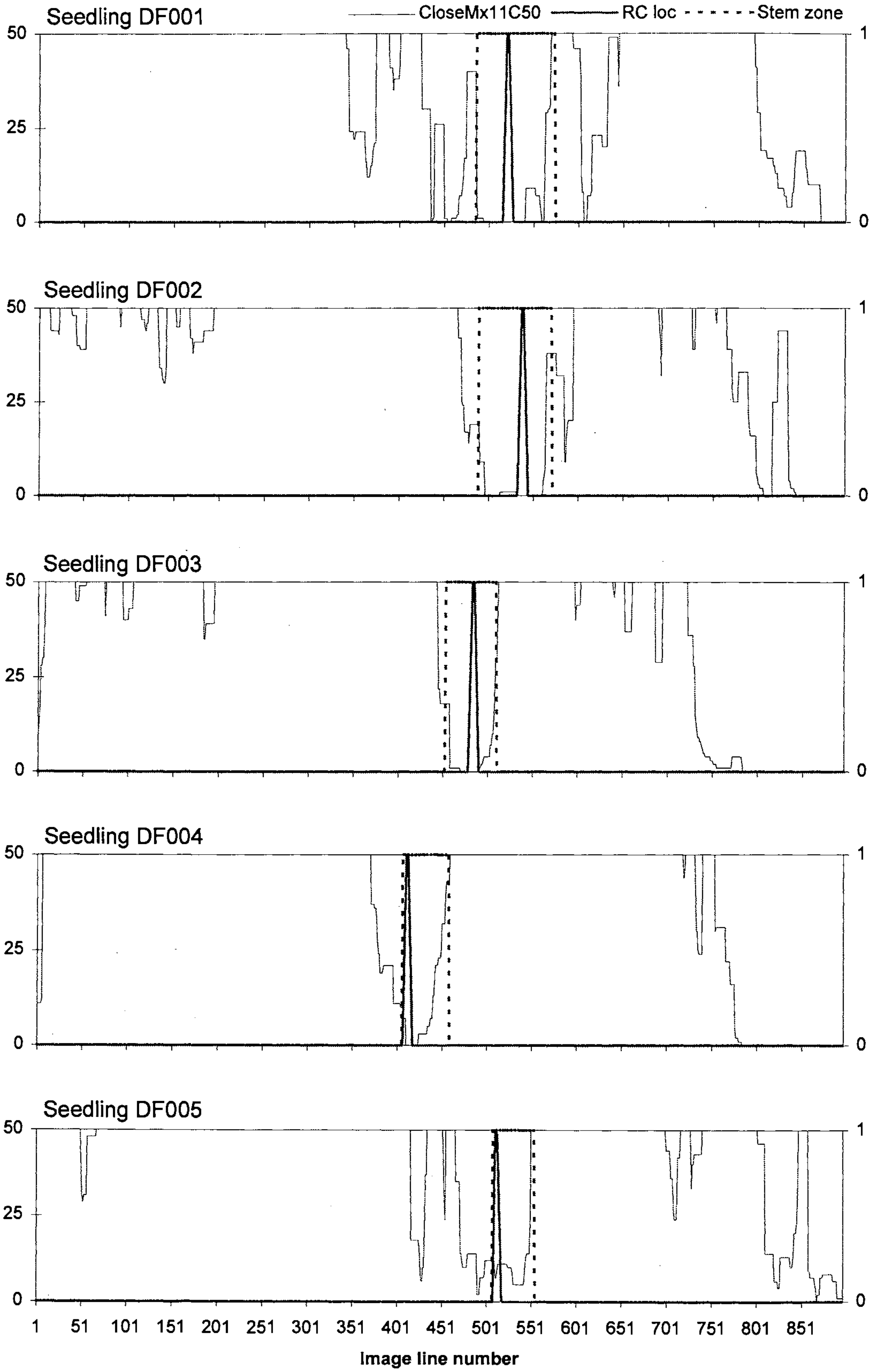


Figure B19. Feature "clipped maximum close area" for 5 seedlings.



APPENDIX C--RESPONSE OF SELECTED NEURAL NETWORKS FOR FIVE SEEDLINGS

Figure C1. Response of network I4a for five seedlings.

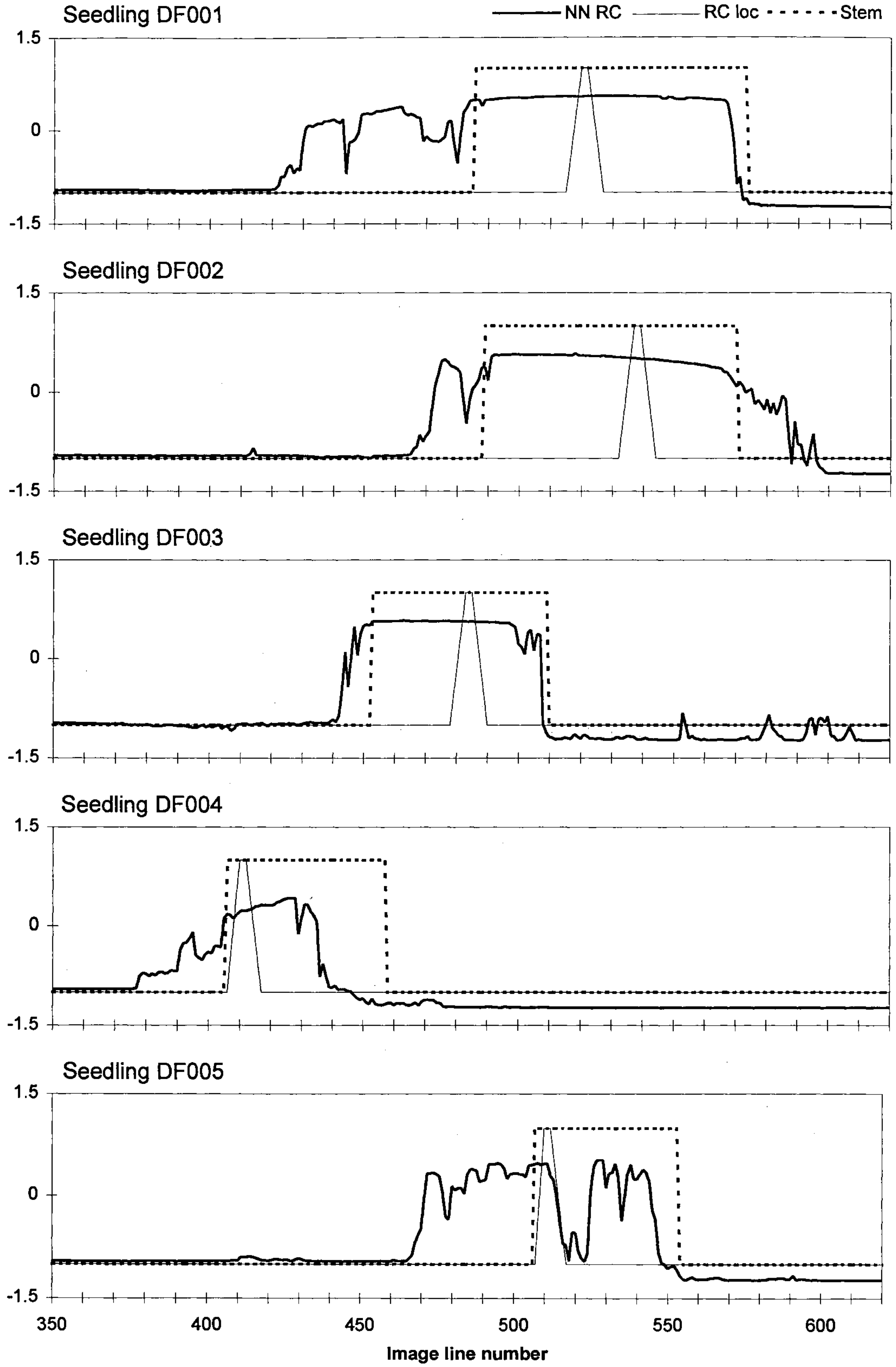


Figure C2. Response of network I5e2 for five seedlings.

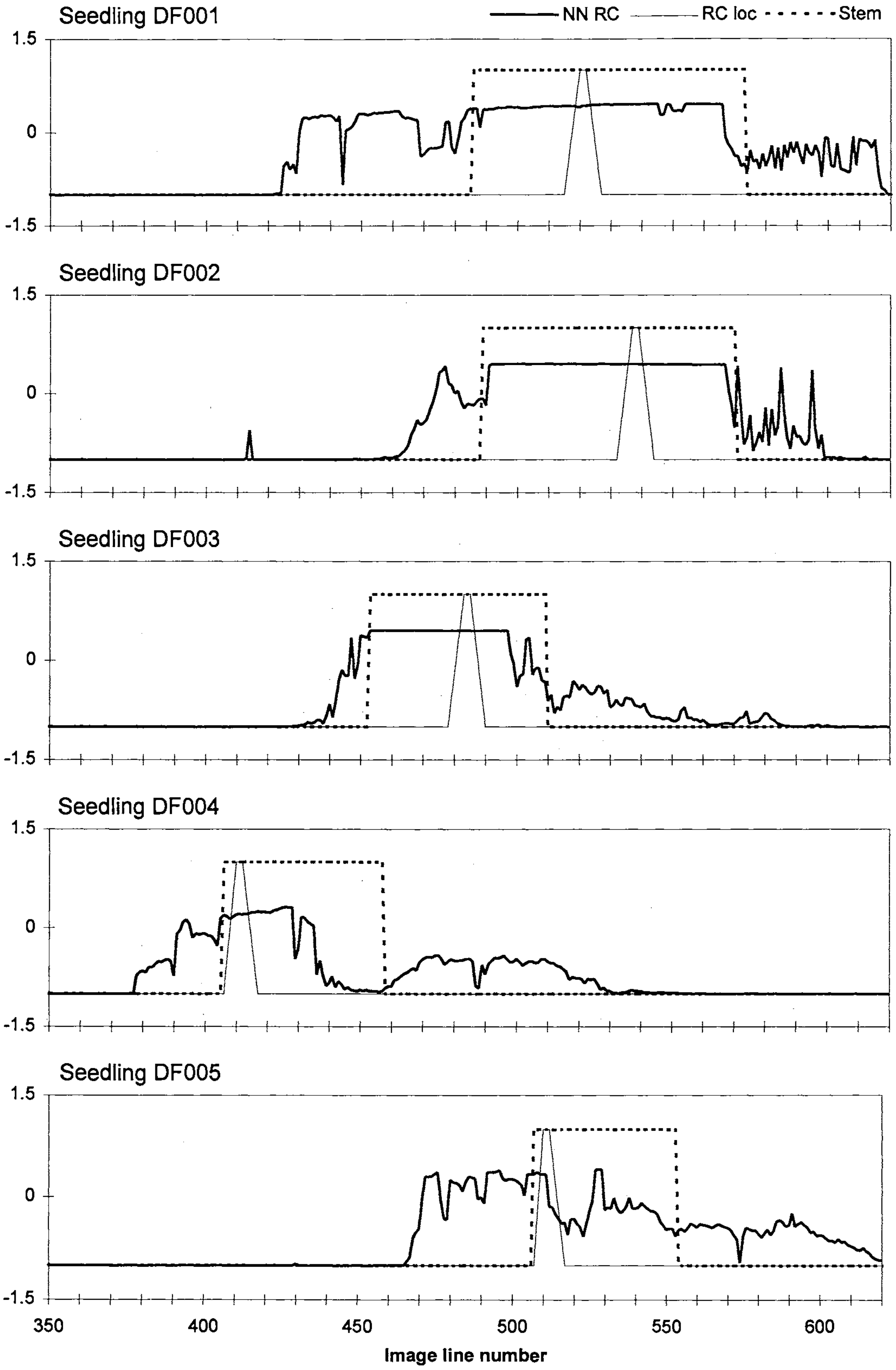


Figure C3. Response of network l7e for five seedlings.

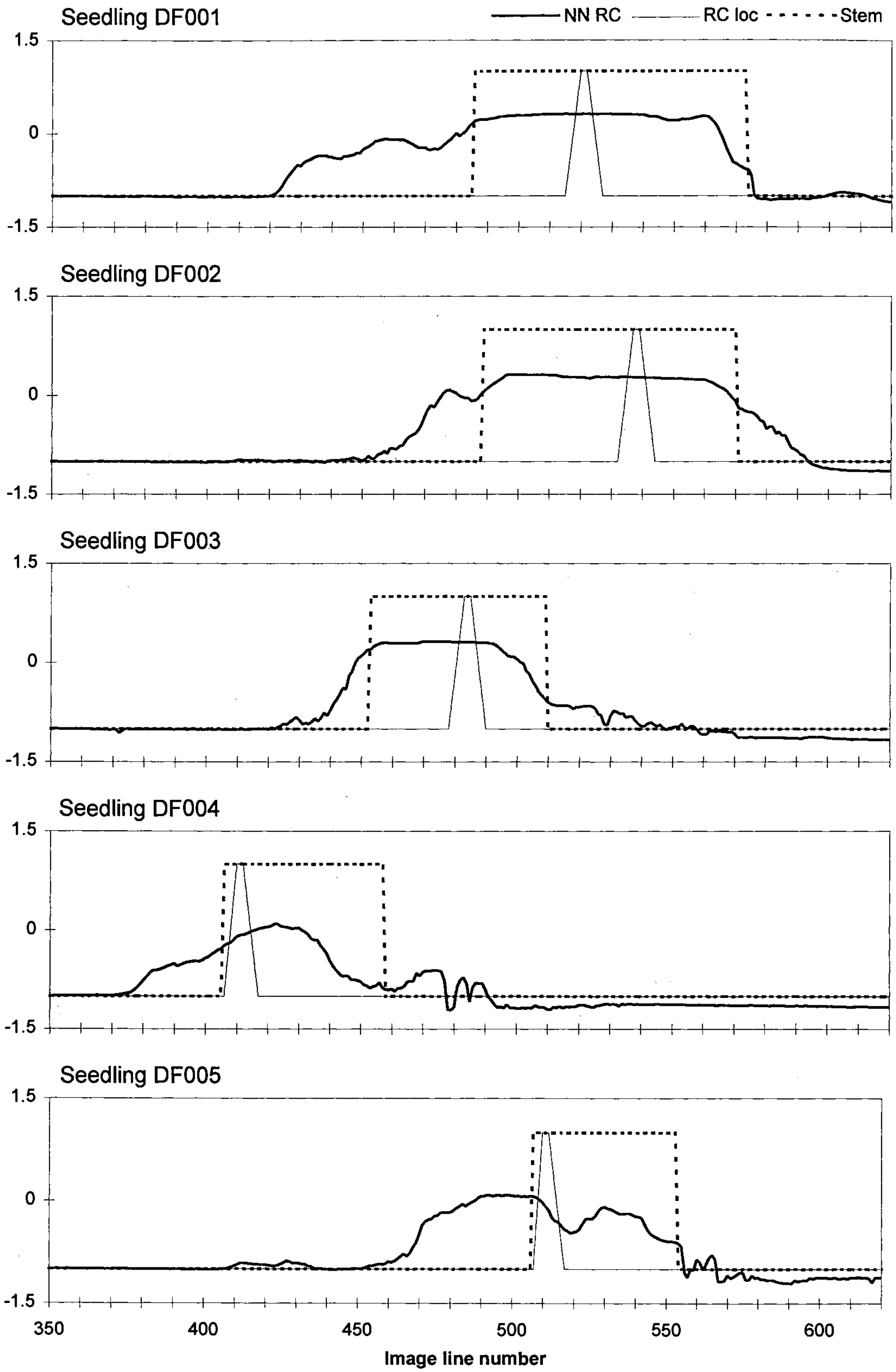


Figure C4. Response of network h5c for five seedlings.

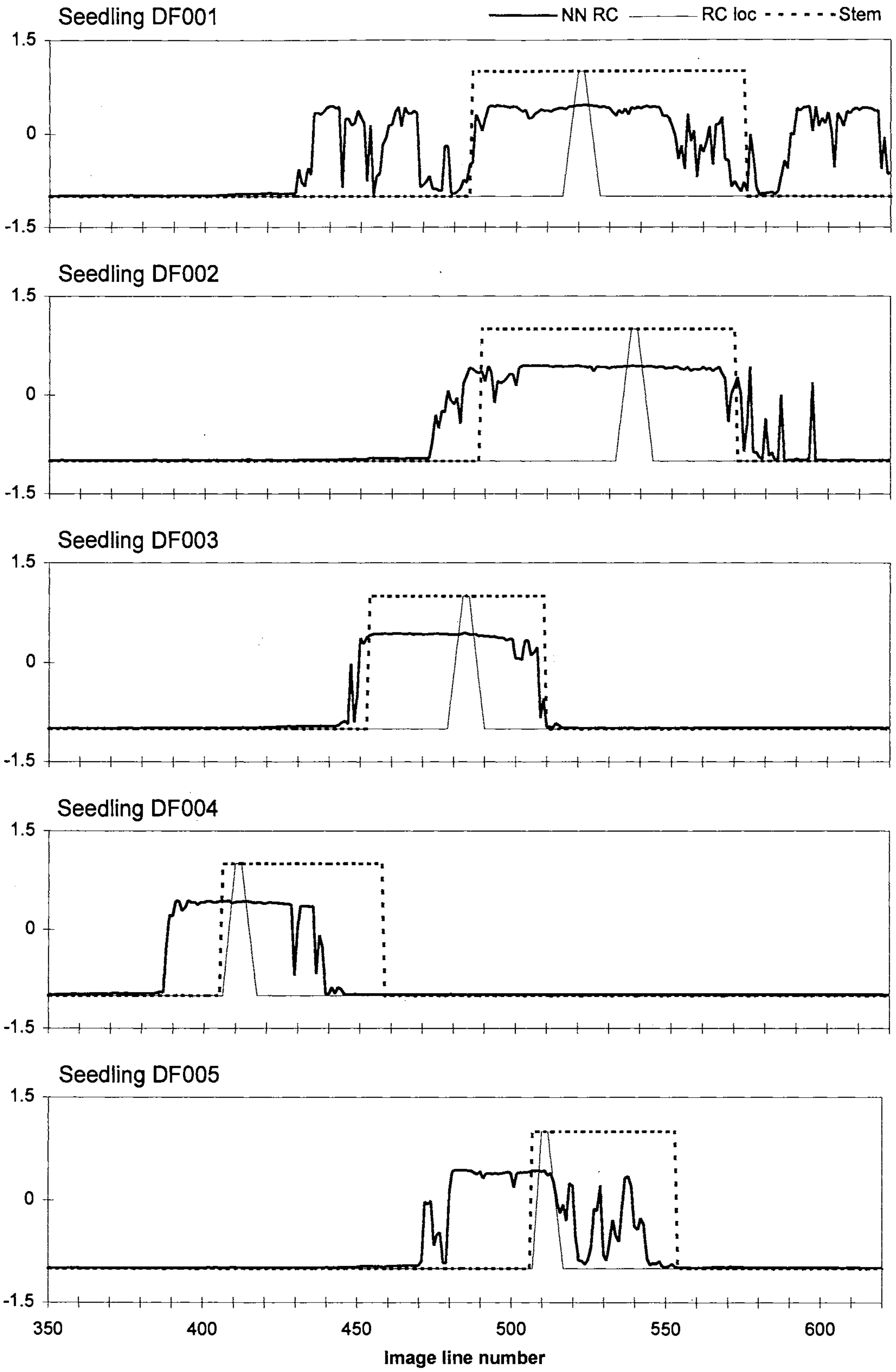


Figure C5. Response of network m4i for five seedlings.

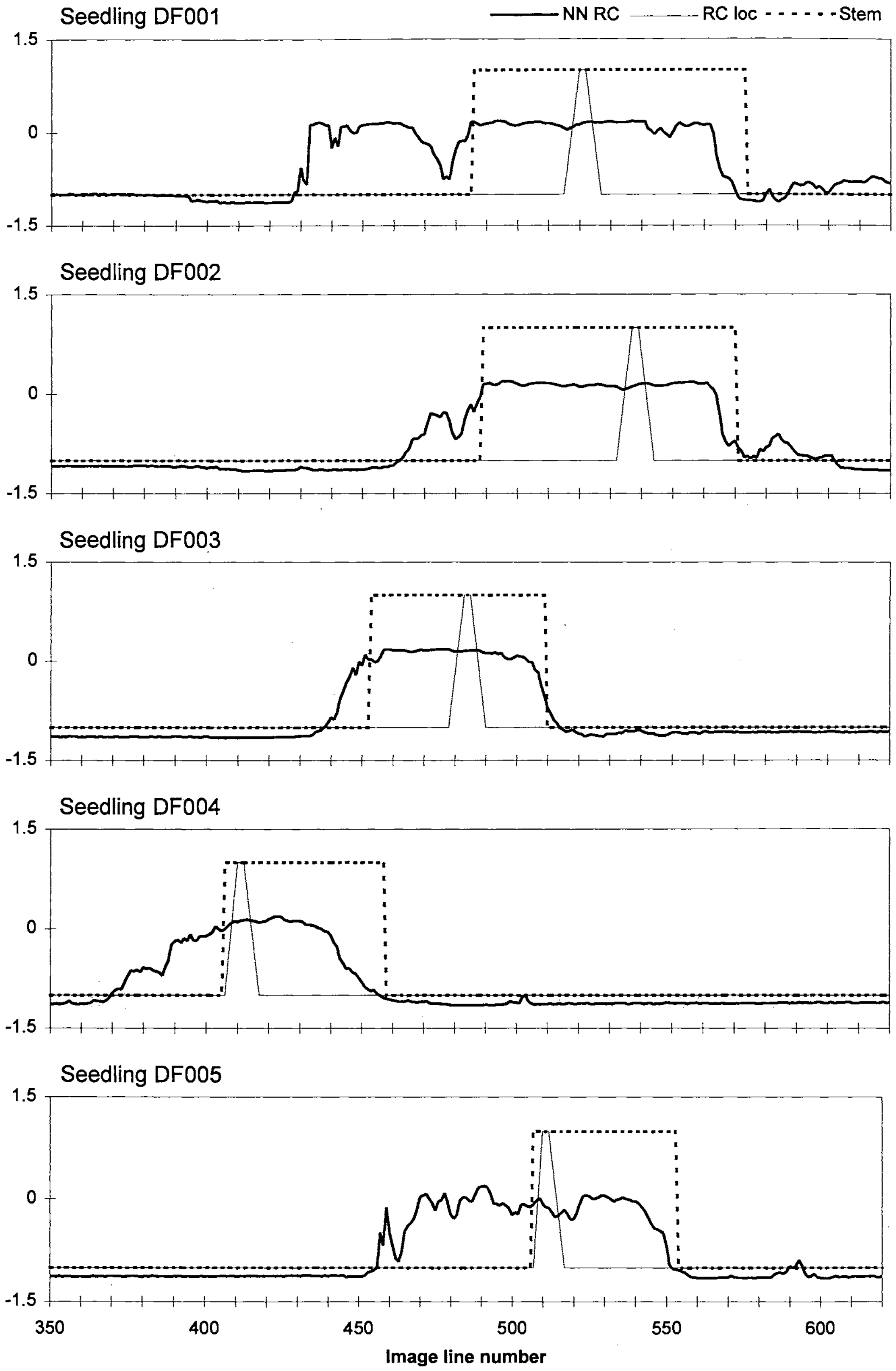


Figure C6. Response of network w15 for five seedlings.

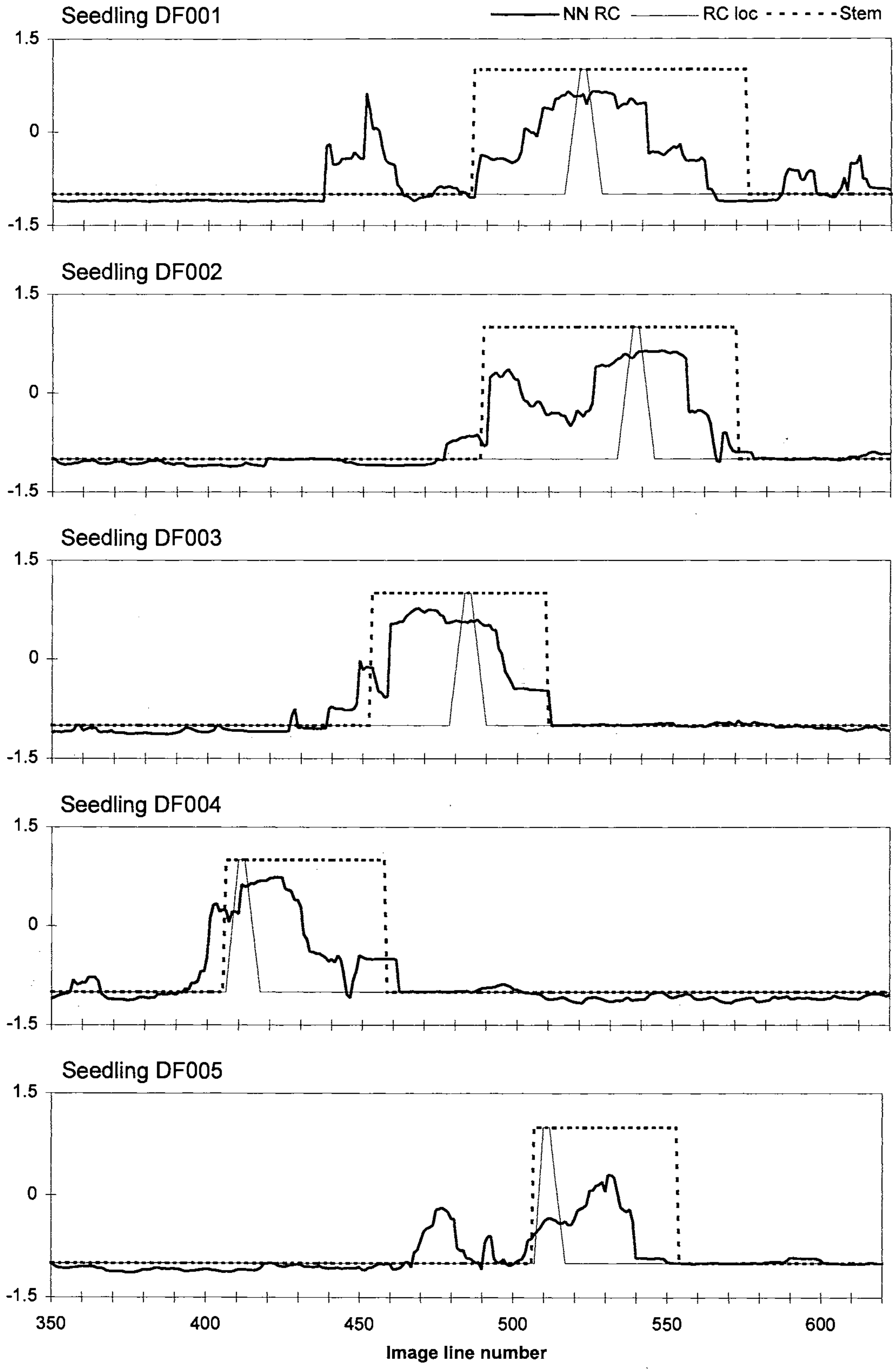
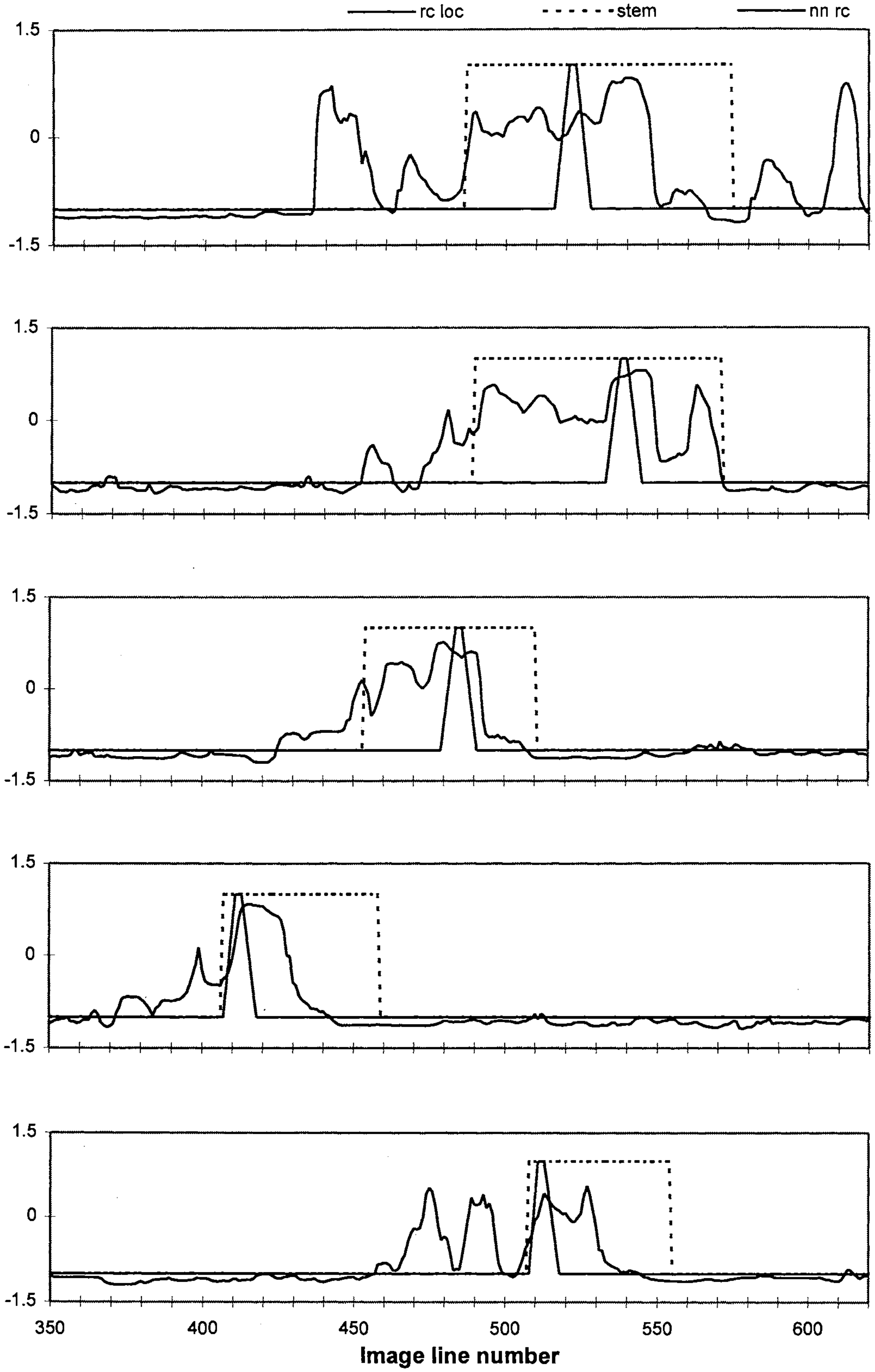


Figure C7. Response of network w7 for five seedlings.



APPENDIX D--SUMMARY OF NEURAL NETWORK FEATURES, ARCHITECTURE, AND PERFORMANCE

Table D1. Composite feature summary for NN1 networks.

NN1 feature	Component features
n	Nline
a	InvRuns, InvArea, InvSpan
b	Runs, InvArea, InvSpan
c	SmRuns, SmArea, SmSpan
d	DelSmRuns, DelSmArea, DelSmSpan
e	M2, M3
f	ConRuns
g	ConArea, CloseArea, UnCon
h	Hist1, Hist2, Hist3, Hist4, Hist5, Hist6, Hist7, Hist8
r	Runs
s	SmRuns
u	NCArea
v	DelNCArea

Table D2a. Summary of NN1 features, architecture, and performance.

	Features							Architecture					Train, 4 Net Average				Test, 4 Net Average				Best Net				
Network	n	a	b	c	d	r	u	Inputs	Hidden 1	Hidden 2	Outputs	Treatment name	RC err	Err stdev	% on stem	found stem	RC err	Err stdev	% on stem	found stem	Net	RC err	Err stdev	% on stem	found stem
l1		y						3	2		1	a201	-20	75	0.79	0.89	57	118	0.70	0.73	1	51	117	0.72	0.74
l1a			y					3	2		1	b201	-4	101	0.76	0.82	61	117	0.70	0.74	3	60	112	0.70	0.76
l1b			y					3	0		1	b001	-11	93	0.76	0.88	46	110	0.67	0.71	2	40	107	0.68	0.72
l1c		y						3	2		4	a204	-5	55	0.79	0.90	53	118	0.69	0.72	4	51	117	0.72	0.74
l1d		y						3	3		4	a304	-10	55	0.80	0.89	56	117	0.69	0.71	4	51	117	0.72	0.74
l1e				y				3	2		4	c204	-3	104	0.75	0.82	65	119	0.69	0.72	2	56	108	0.74	0.78
l1f				y				3	3		4	c304	-5	100	0.77	0.84	64	118	0.70	0.73	4	61	112	0.72	0.76
l1f2				y				3	3	3	4	c334	-1	96	0.74	0.85	67	124	0.63	0.68	1	52	120	0.68	0.70
l1g				y	y			6	3		4	cd304	-14	89	0.81	0.88	31	98	0.73	0.77	2	28	94	0.74	0.80
l1g2				y	y			6	3	3	4	cd334	-9	87	0.78	0.86	43	108	0.70	0.74	3	30	97	0.74	0.78
l1h				y	y			6	4		4	cd404	-10	95	0.80	0.87	38	105	0.73	0.77	1	35	100	0.74	0.78
l1h2				y	y			6	4	4	4	cd444	-4	96	0.79	0.86	46	113	0.72	0.76	1	42	108	0.74	0.78
l4	y	y						4	2		4	na204	0	13	0.90	0.98	12	48	0.79	0.95	3	4	17	0.82	0.98
l4a	y		y					4	2		4	nb204	-2	18	0.84	0.97	6	20	0.75	0.96	1	5	18	0.80	0.98
l4b	y	y						4	3		4	na304	0	12	0.90	0.99	9	35	0.84	0.95	2	3	16	0.86	0.94
l4c	y			y				4	2		4	nc204	2	22	0.79	0.97	9	32	0.51	0.90	1	11	28	0.56	0.92
l4d	y			y				4	3		4	nc304	2	20	0.86	0.97	11	27	0.71	0.92	4	9	25	0.82	0.92
l4da1	y			y				4	3		4	t1nc304	0	23	0.81	0.96	8	28	0.64	0.91	4	6	21	0.76	0.92
l4da2	y			y				4	3		4	t2nc304	6	22	0.84	0.96	14	30	0.70	0.90	4	18	30	0.74	0.92
l4da3	y			y				4	3		4	t3nc304	-3	24	0.78	0.96	5	29	0.59	0.90	3	11	30	0.78	0.90
l4da4	y			y				4	3		4	t4nc304	-5	24	0.75	0.96	1	24	0.63	0.96	4	5	19	0.74	0.96
l4d2	y			y				4	3	3	4	nc334	-1	20	0.83	0.97	7	25	0.68	0.92	2	5	19	0.78	0.96
l4e	y			y	y			7	2		4	ncd204	-1	21	0.78	0.97	7	26	0.53	0.94	4	7	23	0.64	0.94
l4f	y			y	y			7	3		4	ncd304	0	20	0.81	0.97	8	24	0.64	0.94	3	8	23	0.72	0.94
l4f2	y			y	y			7	3	3	4	ncd334	1	19	0.80	0.99	9	25	0.61	0.94	2	10	22	0.66	0.96
l4g	y			y	y			7	4		4	ncd404	1	21	0.84	0.98	8	22	0.72	0.93	1	8	19	0.76	0.94
l4g2	y			y	y			7	4	4	4	ncd444	4	19	0.85	0.99	11	23	0.72	0.93	3	8	19	0.68	0.96
l4h	y					y		2	2		4	nr204	0	13	0.90	0.97	4	17	0.84	0.97	2	3	16	0.84	0.98
l4i	y					y	y	3	3		4	nru304	-9	35	0.82	0.95	1	17	0.86	0.95	4	1	17	0.86	0.96

Table D2b. Summary of NN1 features, architecture, and performance.

	Features								Architecture					Train, 4 Net Average				Test, 4 Net Average				Best Net				
Network	n	a	b	c	d	e	u	v	Inputs	Hidden 1	Hidden 2	Outputs	Treatment name	RC err	Err stdev	% on stem	found stem	RC err	Err stdev	% on stem	found stem	Net	RC err	Err stdev	% on stem	found stem
I2		y				y			5	2		1	ae201	-12	48	0.81	0.91	55	112	0.70	0.74	3	47	105	0.70	0.76
I2a			y			y			5	2		1	be201	-2	105	0.75	0.83	62	116	0.68	0.72	4	59	109	0.70	0.76
I2b			y			y			5	2		4	be204	-2	104	0.77	0.82	63	119	0.68	0.71	1	61	112	0.72	0.74
I2c		y				y			5	2		4	ae204	-15	66	0.80	0.88	42	109	0.74	0.78	1	41	101	0.76	0.80
I2d		y				y			5	3		4	ae304	-8	45	0.82	0.93	45	112	0.72	0.76	1	46	107	0.76	0.78
I5a	y		y			y			6	2		4	nbe204	-1	21	0.87	0.97	5	19	0.77	0.98	4	4	19	0.80	0.98
I5b	y		y			y			6	3		4	nbe304	1	21	0.88	0.96	9	24	0.78	0.92	2	5	19	0.82	0.96
I5c	y		y			y			6	4		4	nbe404	-1	20	0.87	0.96	5	18	0.83	0.97	3	5	17	0.86	0.98
I5d	y	y				y			6	2		4	nae204	-2	15	0.89	0.99	7	30	0.80	0.96	4	2	17	0.80	0.98
I5e	y	y				y			6	3		4	nae304	0	12	0.89	0.99	6	30	0.82	0.94	1	2	17	0.80	0.96
I5e2	y	y				y			6	3	3	4	nae334	-1	13	0.89	0.97	5	16	0.92	0.96	4	5	15	0.94	0.98
I5f	y	y				y			6	4		4	nae404	0	12	0.90	0.98	11	48	0.85	0.94	4	12	42	0.90	0.96
I5f2	y	y				y			6	4	4	4	nae444	-1	15	0.89	0.98	5	16	0.90	0.98	3	6	15	0.94	0.98
I5g	y			y		y			6	3		4	nce304	1	22	0.89	0.95	10	23	0.76	0.94	4	5	18	0.86	0.98
I5g2	y			y		y			6	3	3	4	nce334	3	20	0.89	0.95	10	24	0.71	0.94	1	5	19	0.82	0.98
I5h	y			y		y			6	4		4	nce404	1	22	0.89	0.94	8	21	0.77	0.96	3	4	18	0.80	0.98
I5h2	y			y		y			6	4	4	4	nce444	2	18	0.91	0.95	8	18	0.79	0.97	3	9	15	0.88	0.98
I5i	y			y	y	y			9	3		4	ncde304	1	21	0.86	0.95	8	18	0.78	0.96	2	6	14	0.84	1.00
I5i2	y			y	y	y			9	3	3	4	ncde334	3	16	0.88	0.97	10	21	0.73	0.95	1	8	18	0.80	0.94
I5j	y			y	y	y			9	4		4	ncde404	0	18	0.85	0.96	7	20	0.74	0.96	2	2	16	0.76	0.98
I5j2	y			y	y	y			9	4	4	4	ncde444	2	19	0.87	0.97	9	20	0.71	0.97	4	2	18	0.76	0.96
I5k2	y	y				y	y	y	8	3	3	4	naeuv33	0	14	0.88	1.00	4	16	0.85	0.98	1	6	15	0.94	0.98

Table D2c. Summary of NN1 features, architecture, and performance.

	Features						Architecture					Train, 4 Net Average				Test, 4 Net Average				Best Net				
Network	n	h	s	c	f	g	Inputs	Hidden 1	Hidden 2	Outputs	Treatment name	RC err	Err stdev	% on stem	found stem	RC err	Err stdev	% on stem	found stem	Net	RC err	Err stdev	% on stem	found stem
I3		y					8	2		1	h201	-6	51	0.86	0.86	12	60	0.81	0.87	3	12	54	0.86	0.90
I3a		y					8	2		1	h'201	-17	78	0.86	0.87	11	59	0.82	0.87	1	12	54	0.86	0.90
I3b		y					8	2		4	h204	-4	54	0.85	0.87	4	45	0.76	0.82	2	7	38	0.90	0.94
I3c		y					8	3		4	h304	-4	56	0.87	0.87	16	66	0.81	0.86	2	7	38	0.90	0.94
I3d		y					8	4		4	h404	-4	54	0.86	0.87	12	54	0.88	0.90	3	12	54	0.88	0.90
I6	y	y					9	2		4	nh204	0	42	0.85	0.93	6	38	0.63	0.80	3	3	19	0.82	0.90
I6a	y	y					9	3		4	nh304	0	21	0.88	0.92	7	25	0.74	0.88	2	7	23	0.74	0.90
I6b	y	y					9	4		4	nh404	1	14	0.88	0.96	5	22	0.79	0.90	4	4	16	0.86	0.92
I7	y			y	y	y	8	3		4	ncfg304	2	20	0.85	0.96	10	29	0.71	0.90	4	14	27	0.76	0.92
I7a	y			y	y	y	8	4		4	ncfg404	2	20	0.87	0.94	11	29	0.75	0.90	1	15	28	0.76	0.92
I7b				y	y	y	7	3		4	cfg304	-9	92	0.79	0.85	42	101	0.73	0.77	4	39	93	0.78	0.80
I7c				y	y	y	7	4		4	cfg404	-7	98	0.78	0.84	49	104	0.73	0.76	3	35	91	0.78	0.80
I7d	y		y			y	5	3		4	nsg304	2	19	0.85	0.98	8	26	0.70	0.90	2	7	20	0.78	0.92
I7e	y		y			y	5	4		4	nsg404	3	16	0.89	0.99	7	21	0.79	0.93	3	7	20	0.82	0.94
I7f			y			y	4	3		4	sg304	-10	92	0.82	0.86	55	110	0.70	0.74	1	49	105	0.72	0.76
I7g			y			y	4	2		4	sg204	-9	94	0.80	0.86	54	109	0.71	0.76	2	49	105	0.72	0.76

Table D3. Summary of NN2 features, architecture, and performance.

	NN1					Architecture						Train, 4 Net Average				Test, 4 Net Average				Best Net				
	Net	Features										RC err	Err stdev	% on stem	found stem	RC err	Err stdev	% on stem	found stem	Net	RC err	Err stdev	% on stem	found stem
Network		rc	foli	stem	roots	Inputs	Lines	FOV	Pitch	Hidden 1	Treatment name													
h1	l1a	y				15	15	140	10	3	bC3	-4	37	0.88	0.96	8	32	0.87	0.94	1	6	16	0.90	0.96
h1a	l1a	y				15	15	140	10	5	bC5	-4	47	0.83	0.93	4	28	0.81	0.90	1	3	15	0.86	0.92
h1b	l1a	y				15	15	100	10,5	5	bC5'	-6	45	0.87	0.95	21	54	0.85	0.90	3	20	53	0.86	0.90
h4	l4	y	y		y	45	15	140	10	15	naCFR15	-2	18	0.85	0.94	2	20	0.69	0.94	2	1	18	0.68	0.96
h4a	l4	y	y		y	45	15	140	10	5	naCFR5	1	18	0.86	0.96	4	17	0.71	0.97	1	2	16	0.72	0.98
h4b	l4	y	y			30	15	140	10	15	naCF15	1	15	0.88	0.97	3	18	0.72	0.96	4	0	18	0.68	0.94
h4c	l4	y	y			30	15	140	10	5	naCF5	0	16	0.87	0.97	2	18	0.69	0.94	2	0	18	0.66	0.92
h4d	l4	y	y		y	45	15	140	10	3	naCFR3	1	16	0.87	0.96	2	17	0.72	0.97	1	1	18	0.72	0.92
h4e	l4	y	y			30	15	140	10	3	naCF3	-2	16	0.85	0.97	0	18	0.69	0.96	4	0	17	0.68	0.96
h4f	l4	y	y	y	y	60	15	140	10	5	naCFSR5	-1	15	0.87	0.97	2	18	0.73	0.95	4	0	19	0.70	0.94
h4g	l4	y	y	y	y	60	15	140	10	3	naCFSR3	-1	12	0.87	0.97	1	18	0.70	0.97	4	0	18	0.70	0.96
h4h	l4	y				15	15	140	10	5	naC5	-1	13	0.89	0.98	1	18	0.71	0.97	3	0	18	0.68	0.98
h4i	l4	y	y		y	45	15	140	10	5	naCFR55	2	15	0.90	0.98	3	17	0.71	0.94	1	0	18	0.72	0.90
h4j	l4	y	y		y	45	15	140	10	5	naCFR54	0	13	0.88	0.97	1	18	0.68	0.94	4	1	18	0.68	0.94
h4k	l4	y	y		y	45	15	140	10	5	naCFR53	0	13	0.85	0.96	3	18	0.71	0.97	3	0	19	0.68	0.98
h5	l5b	y	y		y	45	15	140	10	5	nbeCFR5	-1	12	0.87	1.00	11	32	0.65	0.91	2	9	31	0.64	0.92
h5a	l5f	y	y			30	15	140	10	5	naeCF5	0	18	0.87	0.96	4	17	0.77	0.97	4	3	17	0.82	0.96
h5b	l5f	y	y		y	45	15	140	10	3	naeCFR3	-2	12	0.83	1.00	-1	17	0.66	0.98	3	0	18	0.68	0.98
h5c	l5f	y	y			30	15	140	10	3	naeCF3	4	16	0.90	0.98	5	16	0.80	0.98	2	4	16	0.86	0.96
h6	l6a	y	y		y	45	15	140	10	5	nhCFR5	2	18	0.89	0.95	9	27	0.62	0.90	2	6	26	0.54	0.94
h6a	l6a	y	y			30	15	140	10	5	nhCF5	0	20	0.86	0.92	7	26	0.58	0.92	1	5	25	0.56	0.92
h6b	l6a	y				15	15	140	10	5	nhC5	5	16	0.88	0.96	11	26	0.71	0.91	3	7	25	0.60	0.94
h6c	l6a	y	y		y	45	15	140	10	3	nhCFR3	1	20	0.86	0.94	9	27	0.56	0.90	2	3	25	0.54	0.94
h6d	l6a	y	y			30	15	140	10	3	nhCF3	2	18	0.91	0.94	9	27	0.64	0.91	2	5	28	0.54	0.92
h6e	l6a	y				15	15	140	10	3	nhC3	6	15	0.91	0.97	11	27	0.67	0.90	2	10	26	0.62	0.90

Table D4. Composite feature summary for NN3 networks.

NN3 feature	Component featutres	
a	SmArea	
b	ConArea	
c	CloseArea	
d	ConMn41, or ConMn61	
e	ConMn41C50	
f	CloseAreaC50	
a0	ConA11, CloseA11	
a2	ConA*C25, CloseA*C25	* = 5, 11, 15, or 21 depending on pitch
a5	ConA11C50, CloseA11C50	
m2	ConMn*C25, CloseMx*C25	* = 5, 11, 15, or 21 depending on pitch
m5	ConMn11C50, CloseMx11C50	
d0	DelConC100, DelCloseC100	

Table D5a. Summary of wide FOV NN3 features, architecture, and performance.

	Features					Architecture							Train, 4 Net Average				Test, 4 Net Average				Best Net				
Network	a0	a5	a2	m5	m2	Inputs	Lines	Pitch	FOV	Hidden 1	Hidden 2	Treatment name	RC err	Err stdev	% on stem	found stem	RC err	Err stdev	% on stem	found stem	Net	RC err	Err stdev	% on stem	found stem
w1	y					30	15	10	140	5		a01510A	-1	20	0.76	0.90	0	28	0.52	0.75	1	-1	28	0.58	0.74
w1a	y					30	15	10	140	10		a01510B	-4	21	0.70	0.87	-4	29	0.55	0.70	3	3	26	0.68	0.82
w1b	y					30	15	10	140	15		a01510C	2	28	0.60	0.78	1	36	0.45	0.61	2	-2	34	0.54	0.66
w2		y				30	15	10	140	5		a51510A	1	11	0.96	1.00	3	14	0.84	0.97	1	2	12	0.88	0.98
w2a		y				30	15	10	140	10		a51510B	3	10	0.99	1.00	2	15	0.80	0.93	1	2	17	0.78	0.94
w2b		y				30	15	10	140	15		a51510C	2	11	0.99	1.00	3	15	0.85	0.93	1	3	15	0.86	0.92
w2l		y				30	15	10	140	5	3	a51510D	1	9	0.96	1.00	0	16	0.76	0.92	2	0	17	0.74	0.92
w2m		y				30	15	10	140	10	5	a51510E	0	8	0.95	1.00	0	16	0.75	0.92	1	0	16	0.78	0.92
w2n		y				30	15	10	140	15	7	a51510F	0	9	0.93	0.99	-1	16	0.71	0.90	1	-1	15	0.72	0.90
w5			y			30	15	10	140	5		a21510A	1	10	0.98	1.00	2	14	0.88	0.96	2	2	14	0.88	0.96
w5a			y			30	15	10	140	10		a21510B	1	10	0.99	1.00	2	14	0.87	0.96	1	2	12	0.84	0.96
w5b			y			30	15	10	140	15		a21510C	1	10	0.98	1.00	2	13	0.86	0.96	4	2	11	0.88	0.98
w5l			y			30	15	10	140	5	3	a21510D	1	11	0.98	1.00	2	12	0.85	0.96	3	1	9	0.86	0.98
w5m			y			30	15	10	140	10	5	a21510E	2	11	0.97	0.98	4	14	0.87	0.96	3	3	13	0.84	0.94
w5n			y			30	15	10	140	15	7	a21510F	1	7	0.99	1.00	2	16	0.78	0.90	3	1	13	0.80	0.94
w3				y		30	15	10	140	5		m51510A	1	9	0.95	1.00	3	12	0.86	0.98	2	3	12	0.88	0.98
w3a				y		30	15	10	140	10		m51510B	2	9	0.98	1.00	3	13	0.89	0.98	3	3	13	0.88	0.98
w3b				y		30	15	10	140	15		m51510C	3	9	0.98	1.00	3	13	0.86	0.97	3	2	12	0.86	0.96
w3l				y		30	15	10	140	5	3	m51510D	2	8	0.97	1.00	2	14	0.79	0.96	2	2	14	0.80	0.94
w3m				y		30	15	10	140	10	5	m51510E	2	8	0.97	1.00	4	13	0.89	0.96	3	3	13	0.86	0.94
w3n				y		30	15	10	140	15	7	m51510F	2	8	0.97	1.00	6	15	0.85	0.93	2	5	15	0.86	0.96
w4					y	30	15	10	140	5		m21510A	2	9	0.98	1.00	2	12	0.90	0.95	1	2	13	0.90	0.94
w4a					y	30	15	10	140	10		m21510B	1	7	0.98	1.00	4	14	0.85	0.89	4	3	13	0.86	0.90
w4l					y	30	15	10	140	5	3	m21510D	1	8	0.99	1.00	2	12	0.86	0.93	1	1	14	0.86	0.96
w4m					y	30	15	10	140	10	5	m21510E	2	8	0.98	1.00	4	15	0.83	0.88	3	3	14	0.84	0.88

Table D5b. Summary of wide FOV NN3 features, architecture, and performance.

	Features		Architecture							Train, 4 Net Average				Test, 4 Net Average				Best Net				
Network	a2	m2	Inputs	Lines	Pitch	FOV	Hidden 1	Hidden 2	Treatment name	RC err	Err stdev	% on stem	found stem	RC err	Err stdev	% on stem	found stem	Net	RC err	Err stdev	% on stem	found stem
w7	y		30	15	5	70	5		a21505A	2	11	0.97	0.99	2	13	0.92	0.97	2	1	15	0.90	0.94
w7a	y		30	15	5	70	10		a21505B	0	13	0.92	0.97	1	16	0.85	0.94	2	1	14	0.88	0.96
w7b	y		30	15	5	70	15		a21505C	-2	13	0.90	0.96	1	18	0.83	0.89	2	0	18	0.84	0.90
w7l	y		30	15	5	70	5	3	a21505D	0	12	0.96	1.00	2	14	0.85	0.95	1	3	12	0.86	0.96
w7m	y		30	15	5	70	10	5	a21505E	-1	11	0.92	0.97	0	19	0.78	0.88	1	0	20	0.76	0.86
w7n	y		30	15	5	70	15	7	a21505F	0	13	0.92	0.96	2	20	0.77	0.85	1	1	20	0.74	0.88
w6		y	30	15	5	70	5		m21505A	2	11	0.99	1.00	2	14	0.89	0.96	1	1	13	0.88	0.96
w6a		y	30	15	5	70	10		m21505B	0	10	0.99	0.99	2	14	0.89	0.96	3	1	14	0.84	0.96
w6l		y	30	15	5	70	5	3	m21505D	1	11	0.99	1.00	2	15	0.88	0.93	2	1	15	0.88	0.92
w6m		y	30	15	5	70	10	5	m21505E	1	9	0.97	0.99	0	19	0.78	0.85	2	0	17	0.80	0.90
w8	y		30	15	15	210	5		a21515A	0	13	0.92	0.98	-1	13	0.83	0.96	4	0	14	0.82	0.94
w8a	y		30	15	15	210	10		a21515B	1	12	0.98	1.00	1	14	0.80	0.96	1	0	13	0.82	0.98
w8l	y		30	15	15	210	5	3	a21515D	2	11	0.94	1.00	1	14	0.87	0.96	2	1	13	0.88	0.96
w8m	y		30	15	15	210	10	5	a21515E	1	9	0.98	0.99	1	14	0.80	0.91	3	0	14	0.80	0.90
w9		y	30	15	15	210	5		m21515A	-1	10	0.95	1.00	0	14	0.86	0.95	4	-1	14	0.84	0.94
w9a		y	30	15	15	210	10		m21515B	1	10	0.98	1.00	2	13	0.86	0.96	3	0	14	0.82	0.94
w9l		y	30	15	15	210	5	3	m21515D	-1	10	0.95	0.99	0	16	0.80	0.95	1	-1	14	0.80	0.96
w9m		y	30	15	15	210	10	5	m21515E	0	9	0.96	0.99	0	15	0.79	0.90	1	0	16	0.80	0.90
w10	y		30	15	20	280	5		a21520A	2	12	0.96	1.00	2	13	0.77	0.95	1	2	13	0.78	0.96
w10a	y		30	15	20	280	10		a21520B	1	11	0.97	1.00	2	14	0.77	0.96	1	1	13	0.74	0.96
w10l	y		30	15	20	280	5	3	a21520D	1	13	0.90	0.99	1	13	0.75	0.92	4	0	13	0.74	0.92
w10	y		30	15	20	280	10	5	a21520E	0	9	0.96	1.00	4	13	0.79	0.98	1	2	13	0.70	0.98
w11		y	30	15	20	280	5		m21520A	2	11	0.97	0.99	4	13	0.86	0.94	3	4	11	0.88	0.94
w11a		y	30	15	20	280	10		m21520B	1	9	0.99	1.00	3	13	0.87	0.95	4	2	13	0.88	0.96
w11l		y	30	15	20	280	5	3	m21520D	1	10	0.99	1.00	4	13	0.86	0.94	4	3	12	0.86	0.94
w11m		y	30	15	20	280	10	5	m21520E	1	9	0.99	1.00	5	13	0.88	0.93	3	4	13	0.86	0.96

Table D5c. Summary of wide FOV NN3 features, architecture, and performance.

	Features			Architecture							Train, 4 Net Average				Test, 4 Net Average				Best Net				
Network	a2	m2	d	Inputs	Lines	Pitch	FOV	Hidden 1	Hidden 2	Treatment name	RC err	Err stdev	% on stem	found stem	RC err	Err stdev	% on stem	found stem	Net	RC err	Err stdev	% on stem	found stem
w12	y			42	21	5	100	5		a22105A	2	11	0.95	0.99	2	14	0.90	0.95	3	1	12	0.90	0.98
w12a	y			42	21	5	100	10		a22105B	1	11	0.93	0.98	3	15	0.83	0.93	4	2	14	0.80	0.94
w12l	y			42	21	5	100	5	3	a22105D	0	13	0.93	0.98	2	15	0.85	0.95	4	1	14	0.92	0.96
w12m	y			42	21	5	100	10	5	a22105E	1	10	0.97	1.00	3	14	0.80	0.94	1	3	12	0.88	0.94
w13		y		42	21	5	100	5		m22105A	2	11	0.98	1.00	3	12	0.92	0.98	2	3	12	0.92	0.98
w13a		y		42	21	5	100	10		m22105B	3	10	1.00	1.00	4	13	0.88	0.96	1	3	11	0.90	0.96
w13l		y		42	21	5	100	5	3	m22105D	1	11	0.96	0.99	4	14	0.89	0.95	3	4	15	0.86	0.92
w13m		y		42	21	5	100	10	5	m22105E	2	9	0.98	1.00	2	15	0.84	0.90	4	0	16	0.80	0.86
w14	y			18	9	10	80	3		a20910A	1	13	0.97	1.00	-1	12	0.80	0.91	1	-1	12	0.82	0.92
w14a	y			18	9	10	80	6		a20910B	1	11	1.00	1.00	-1	14	0.80	0.88	4	0	13	0.84	0.92
w14l	y			18	9	10	80	3	2	a20910D	0	11	0.99	0.99	0	13	0.87	0.91	1	0	13	0.90	0.92
w14m	y			18	9	10	80	6	3	a20910E	2	10	0.98	0.99	1	15	0.83	0.89	3	1	13	0.84	0.88
w15		y		18	9	10	80	3		m20910A	1	10	0.97	1.00	1	10	0.92	0.97	1	1	10	0.92	0.98
w15a		y		18	9	10	80	6		m20910B	2	11	0.98	1.00	3	11	0.93	0.96	1	0	12	0.90	0.96
w15l		y		18	9	10	80	3	2	m20910D	0	12	0.95	1.00	3	12	0.90	0.96	3	0	12	0.88	0.94
w15m		y		18	9	10	80	6	3	m20910E	2	10	0.97	0.99	6	15	0.94	0.94	3	5	12	0.96	0.98
w16	y		y	60	15	10	140	5		a2d1510A	2	10	0.97	1.00	1	13	0.80	0.95	4	1	12	0.78	0.96
w16a	y		y	60	15	10	140	10		a2d1510B	2	9	0.97	1.00	2	13	0.87	0.96	4	1	13	0.82	0.94
w16l	y		y	60	15	10	140	5	3	a2d1510D	1	10	0.96	1.00	2	13	0.84	0.95	4	1	14	0.84	0.94
w16m	y		y	60	15	10	140	10	5	a2d1510E	1	8	0.99	1.00	1	14	0.79	0.92	1	1	12	0.78	0.92
w17		y	y	60	15	10	140	5		m2d1510	2	8	0.98	1.00	3	12	0.90	0.96	3	2	12	0.88	0.96
w17a		y	y	60	15	10	140	10		m2d1510	0	7	0.99	1.00	2	13	0.87	0.93	2	1	13	0.84	0.94
w17l		y	y	60	15	10	140	5	3	2d1510	1	9	0.95	1.00	0	13	0.83	0.92	1	0	12	0.84	0.94
w17m		y	y	60	15	10	140	10	5	m2d1510	1	7	0.97	1.00	1	14	0.80	0.93	3	0	13	0.78	0.92

Table D5d. Summary of narrow FOV NN3 features, architecture, and performance.

	Features			Architecture							Train, 4 Net Average				Test, 4 Net Average				Best Net				
Network	a	b	c	Inputs	Lines	Pitch	FOV	Hidden 1	Hidden 2	Treatment name	RC err	Err stdev	% on stem	found stem	RC err	Err stdev	% on stem	found stem	Net	RC err	Err stdev	% on stem	found stem
m1	y	y	y	36	15	1	14	7		abc107	-5	16	0.85	0.95	-4	15	0.78	0.95	3	-2	13	0.84	0.96
m1a	y	y	y	36	15	1	14	14		abc114	-3	16	0.87	0.94	-2	13	0.83	0.96	1	-2	13	0.84	0.96
m1b	y	y	y	36	15	1	14	21		abc121	-4	16	0.86	0.94	-2	13	0.83	0.96	3	-1	13	0.84	0.96
m1c		y	y	30	15	1	14	6		bc106	-1	11	0.92	1.00	2	13	0.92	0.96	2	1	13	0.92	0.96
m1d		y	y	30	15	1	14	12		bc112	0	11	0.89	1.00	2	14	0.91	0.97	3	0	13	0.90	0.98
m1e		y	y	30	15	1	14	18		bc118	-1	11	0.90	1.00	1	14	0.90	0.97	3	0	14	0.92	0.96
m1f		y	y	30	15	1	14	6	3	bc1063	1	11	0.91	0.99	4	14	0.92	0.96	1	2	13	0.92	0.96
m1g		y	y	30	15	1	14	12	4	bc1124	0	11	0.92	0.99	3	13	0.92	0.97	2	4	12	0.94	0.98
m1h		y	y	30	15	1	14	18	6	bc1186	0	11	0.92	1.00	3	14	0.91	0.96	4	3	14	0.90	0.96
m1i		y	y	30	15	1	14	10		bc1B	0	11	0.90	1.00	1	14	0.90	0.96	1	2	14	0.92	0.96
m2	y	y	y	36	15	2	28	7		abc207	-5	17	0.84	0.94	-3	14	0.81	0.96	2	-3	13	0.82	0.96
m2a	y	y	y	36	15	2	28	14		abc214	-4	16	0.86	0.95	-2	12	0.83	0.97	4	-1	11	0.86	0.98
m2b	y	y	y	36	15	2	28	21		abc221	-4	16	0.87	0.95	-2	12	0.83	0.97	2	-1	10	0.84	0.98
m2c		y	y	30	15	2	28	6		bc206	-6	11	0.81	0.99	-3	12	0.84	0.98	4	-1	13	0.84	0.98
m2d		y	y	30	15	2	28	12		bc212	-5	11	0.83	1.00	-2	13	0.85	0.98	1	0	13	0.88	0.98
m2e		y	y	30	15	2	28	18		bc218	-5	12	0.83	1.00	-1	13	0.83	0.98	4	-1	13	0.84	0.98
m2f		y	y	30	15	2	28	6	3	bc2063	-3	11	0.87	0.99	0	13	0.88	0.97	1	0	13	0.88	0.98
m2g		y	y	30	15	2	28	12	4	bc2124	-4	11	0.87	0.98	-1	13	0.87	0.98	1	0	12	0.88	0.98
m2h		y	y	30	15	2	28	18	6	bc2186	-4	11	0.86	0.99	-1	13	0.87	0.98	2	0	13	0.88	0.98
m3	y	y	y	36	15	3	43	7		abc307	-5	17	0.84	0.94	-3	13	0.79	0.95	2	-2	12	0.82	0.96
m3a	y	y	y	36	15	3	43	14		abc314	-3	15	0.87	0.96	-2	12	0.82	0.97	4	-1	11	0.86	0.98
m3b	y	y	y	36	15	3	43	21		abc321	-3	16	0.85	0.95	-2	12	0.83	0.97	3	-1	11	0.84	0.98
m3c		y	y	30	15	3	43	6		bc306	-11	11	0.68	1.00	-7	11	0.65	1.00	4	-3	13	0.70	0.98
m3d		y	y	30	15	3	43	12		bc312	-7	11	0.77	1.00	-4	13	0.72	0.98	1	-2	14	0.80	0.96
m3e		y	y	30	15	3	43	18		bc318	-8	11	0.76	1.00	-5	12	0.70	0.98	2	-3	13	0.74	0.96
m3f		y	y	30	15	3	43	6	3	bc3063	-6	11	0.83	0.99	-3	14	0.80	0.97	4	-1	13	0.86	0.98
m3g		y	y	30	15	3	43	12	4	bc3124	-7	11	0.81	0.98	-4	13	0.77	0.97	4	-2	14	0.84	0.96
m3h		y	y	30	15	3	43	18	6	bc3186	-7	11	0.79	1.00	-4	13	0.78	0.98	3	-3	13	0.84	0.98

Table D5e. Summary of narrow FOV NN3 features, architecture, and performance.

	Features			Architecture							Train, 4 Net Average				Test, 4 Net Average				Best Net				
Network	a	c	d	Inputs	Lines	Pitch	FOV	Hidden 1	Hidden 2	Treatment name	RC err	Err stdev	% on stem	found stem	RC err	Err stdev	% on stem	found stem	Net	RC err	Err stdev	% on stem	found stem
m4	y	y	y	36	15	1	14	7		acd107	-5	17	0.83	0.95	-3	14	0.81	0.96	3	-2	13	0.84	0.96
m4a	y	y	y	36	15	1	14	14		acd114	-3	16	0.86	0.94	-2	13	0.84	0.96	4	-1	13	0.86	0.96
m4b	y	y	y	36	15	1	14	21		acd121	-4	16	0.85	0.94	-3	14	0.83	0.96	1	-2	13	0.84	0.96
m4c		y	y	30	15	1	14	6		cd106	-2	11	0.92	1.00	1	13	0.93	0.96	2	1	13	0.94	0.96
m4d		y	y	30	15	1	14	12		cd112	-1	11	0.92	1.00	2	13	0.92	0.96	3	0	13	0.92	0.96
m4e		y	y	30	15	1	14	18		cd118	-2	11	0.91	1.00	1	13	0.92	0.96	3	0	13	0.92	0.96
m4f		y	y	30	15	1	14	6	3	cd1063	0	12	0.90	0.99	4	13	0.93	0.97	1	3	12	0.94	0.98
m4g		y	y	30	15	1	14	12	4	cd1124	-1	11	0.88	1.00	2	13	0.87	0.98	2	1	13	0.74	1.00
m4h		y	y	30	15	1	14	18	6	cd1186	-2	12	0.88	1.00	3	12	0.89	0.99	4	2	13	0.92	0.96
m4i		y	y	30	15	1	14	5		cd1A	-1	11	0.90	1.00	2	12	0.93	0.99	3	1	12	0.92	0.98
m4j		y	y	30	15	1	14	10		cd1B	-2	12	0.88	1.00	2	12	0.92	0.99	2	1	11	0.90	1.00
m4k		y	y	30	15	1	14	15		cd1C	-1	11	0.88	1.00	1	13	0.89	0.97	3	1	13	0.82	1.00
m5	y	y	y	36	15	2	28	7		acd207	-5	17	0.84	0.95	-4	14	0.79	0.96	4	-2	12	0.80	0.96
m5a	y	y	y	36	15	2	28	14		acd214	-5	11	0.84	1.00	-1	12	0.85	0.99	2	1	13	0.94	0.98
m5b	y	y	y	36	15	2	28	21		acd221	-5	12	0.84	1.00	-2	12	0.87	0.98	2	0	13	0.88	0.98
m5c		y	y	30	15	2	28	6		cd206	-7	10	0.81	1.00	-4	12	0.85	0.98	1	-3	12	0.86	0.98
m5d		y	y	30	15	2	28	12		cd212	-6	11	0.83	1.00	-3	12	0.87	0.98	1	-2	13	0.90	0.98
m5e		y	y	30	15	2	28	18		cd218	-6	11	0.84	1.00	-3	12	0.86	0.98	4	-3	12	0.86	0.98
m5f		y	y	30	15	2	28	6	3	cd2063	-5	12	0.83	1.00	-1	14	0.80	0.99	4	0	13	0.92	0.98
m5g		y	y	30	15	2	28	12	4	cd2124	-5	11	0.86	0.99	-1	13	0.85	0.99	2	0	13	0.92	0.98
m5h		y	y	30	15	2	28	18	6	cd2186	-3	11	0.90	1.00	-1	13	0.89	0.97	2	0	13	0.90	0.98
m6	y	y	y	36	15	3	42	7		acd307	-4	16	0.86	0.95	-3	13	0.83	0.98	2	-2	11	0.88	1.00
m6a	y	y	y	36	15	3	42	14		acd314	-3	15	0.86	0.95	-2	12	0.83	0.97	2	-1	11	0.84	0.98
m6b	y	y	y	36	15	3	42	21		acd321	-3	16	0.86	0.94	-3	12	0.82	0.97	3	-2	11	0.84	0.98
m6c		y	y	30	15	3	42	6		cd306	-10	11	0.70	1.00	-6	12	0.68	1.00	4	-5	11	0.68	1.00
m6d		y	y	30	15	3	42	12		cd312	-9	11	0.74	1.00	-5	11	0.70	1.00	4	-4	11	0.70	1.00
m6e		y	y	30	15	3	42	18		cd318	-9	11	0.74	1.00	-5	11	0.71	1.00	1	-4	11	0.74	1.00
m6f		y	y	30	15	3	42	6	3	cd3063	-5	12	0.87	0.99	-3	13	0.85	0.98	2	-1	13	0.88	0.98
m6g		y	y	30	15	3	42	12	4	cd3124	-8	13	0.79	0.99	-5	13	0.76	0.98	1	-3	13	0.86	0.98
m6h		y	y	30	15	3	42	18	6	cd3186	-6	12	0.85	0.99	-3	13	0.84	0.98	2	-2	13	0.86	0.98

Table D5f. Summary of narrow FOV NN3 features, architecture, and performance.

	Features			Architecture							Train, 4 Net Average				Test, 4 Net Average				Best Net				
Network	c	e	f	Inputs	Lines	Pitch	FOV	Hidden 1	Hidden 2	Treatment name	RC err	Err stdev	% on stem	found stem	RC err	Err stdev	% on stem	found stem	Net	RC err	Err stdev	% on stem	found stem
m7c	y	y		30	15	1	14	6		ce106	-4	20	0.87	0.89	0	20	0.87	0.88	1	-1	20	0.86	0.88
m7d	y	y		30	15	1	14	12		ce112	-4	19	0.87	0.89	1	20	0.86	0.88	3	0	20	0.86	0.88
m7e	y	y		30	15	1	14	18		ce118	-1	18	0.91	0.91	0	20	0.87	0.88	4	0	20	0.88	0.88
m7i	y	y		30	15	1	14	5		ce1A	-3	17	0.95	0.95	-1	14	0.90	0.92	2	0	12	0.92	0.92
m7j	y	y		30	15	1	14	10		ce1B	0	17	0.93	0.94	1	13	0.93	0.94	2	-1	15	0.92	0.92
m7k	y	y		30	15	1	14	15		ce1C	-3	15	0.96	0.96	2	12	0.94	0.96	4	3	12	0.94	0.96
m7l	y	y		30	15	1	14	5	3	ce1D	0	19	0.88	0.90	1	17	0.91	0.92	3	0	17	0.90	0.92
m7m	y	y		30	15	1	14	10	5	ce1E	0	17	0.93	0.93	0	17	0.89	0.90	2	0	16	0.92	0.94
m7n	y	y		30	15	1	14	15	7	ce1F	2	16	0.95	0.95	0	17	0.89	0.90	3	-1	20	0.86	0.86
m8c		y	y	30	15	1	14	6		ef106	-1	14	0.95	0.96	0	14	0.92	0.95	3	0	13	0.92	0.96
m8d		y	y	30	15	1	14	12		ef112	-1	13	0.96	0.96	1	14	0.94	0.96	2	1	13	0.96	0.96
m8e		y	y	30	15	1	14	18		ef118	-1	13	0.96	0.96	1	14	0.94	0.96	2	1	13	0.96	0.96
m8i		y	y	30	15	1	14	5		ef1A	0	14	0.97	0.97	-1	12	0.95	0.96	3	0	13	0.94	0.96
m8j		y	y	30	15	1	14	10		ef1B	0	13	0.98	0.98	1	12	0.94	0.97	2	0	12	0.94	0.96
m8k		y	y	30	15	1	14	15		ef1C	0	14	0.96	0.97	-1	12	0.92	0.96	1	0	12	0.94	0.96
m8l		y	y	30	15	1	14	5	3	ef1D	2	13	0.97	0.99	0	14	0.91	0.94	3	1	13	0.92	0.94
m8m		y	y	30	15	1	14	10	5	ef1E	2	14	0.94	0.96	0	13	0.91	0.95	3	0	13	0.90	0.96
m8n		y	y	30	15	1	14	15	7	ef1F	2	14	0.95	0.96	1	13	0.94	0.96	3	0	13	0.92	0.96

APPENDIX E--STATISTICAL ANALYSIS OF NEURAL NETWORK PERFORMANCE

Table E1. Statistical comparison of NN1 networks with 1 and 4 outputs.

Features	a		ae		be		h	
Outputs	1	4	1	4	1	4	1	4
Network	l1	l1c	l2	l2c	l2a	l2b	l3	l3b
Mean error, mm	**50.6**	50.9	47.1	**40.9**	**58.9**	61.2	11.8	**7.1**
Stdev error, mm	**117.2**	117.4	105.1	**100.6**	**108.7**	112.0	53.7	**38.5**

	a	ae	be	h
Mean difference	-0.2	6.2	-2.3	4.7
t-test	0.288	0.624	0.719	0.390
F-test	0.991	0.759	0.835	**0.021**

Table E2. Comparison of objective functions used for NN training.

Features	nc				
Network	l4d	l4da1	l4da2	l4da4	l4da3
Mean error, mm	**0.1**	0.8	1.1	-1.0	-4.5
Stdev error, mm	**17.9**	22.5	22.8	25.0	24.5

t-test		l4da1	l4da2	l4da4	l4da3
	l4d	0.714	0.624	0.641	**0.032**
	l4da1		0.689	**0.050**	**0.000**
	l4da2			**0.058**	**0.000**
	l4da4				**0.001**

F-test		l4da1	l4da2	l4da4	l4da3
	l4d	0.113	**0.094**	**0.021**	**0.029**
	l4da1		0.927	0.460	0.540
	l4da2			0.518	0.602
	l4da4				0.899

Objective function	
l4d	Classification rate
l4da1	RC confusion matrix, trapezoidal fuzzy mbr. func.
l4da2	Stem confusion matrix
l4da4	RC confusion matrix, rectangular fuzzy mbr. func.
l4da3	rms error

Table E3a. Statistical comparison of input features for NN1 networks with similar architecture

Architecture	2 hidden layer nodes, 4 outputs													
Features	nae	nh	nr	na	nbe	nb	ncd	nc	h	ae	sg	a	c	be
Network	l5d	l6	l4h	l4	l5a	l4a	l4e	l4c	l3b	l2c	l7g	l1c	l1e	l2b
Mean error, mm	**2.3**	2.8	3.4	3.5	4.5	4.9	7.4	10.7	7.1	40.9	49.0	50.9	55.7	61.2
Stdev error, mm	16.6	19.3	**16.2**	16.7	19.4	18.1	22.9	28.3	38.5	100.6	105.0	117.4	108.1	112.0

t-test		nh	nr	na	nbe	nb	ncd	nc	h	ae	sg	a	c	be
	nae	0.754	**0.100**	**0.020**	**0.068**	**0.003**	**0.005**	**0.006**	0.355	**0.007**	**0.003**	**0.004**	**0.001**	**0.000**
	nh		0.688	0.629	0.442	0.275	**0.078**	**0.028**	0.405	**0.012**	**0.004**	**0.007**	**0.002**	**0.001**
	nr			0.774	0.425	0.159	**0.044**	**0.022**	0.481	**0.010**	**0.003**	**0.005**	**0.001**	**0.001**
	na				0.425	0.158	**0.030**	**0.018**	0.500	**0.010**	**0.003**	**0.005**	**0.001**	**0.001**
	nbe					0.617	**0.027**	**0.029**	0.631	**0.011**	**0.004**	**0.006**	**0.001**	**0.001**
	nb						**0.079**	**0.048**	0.675	**0.013**	**0.004**	**0.007**	**0.002**	**0.001**
	ncd							0.136	0.959	**0.019**	**0.007**	**0.009**	**0.003**	**0.001**
	nc								0.580	**0.031**	**0.011**	**0.015**	**0.004**	**0.002**
	h									**0.013**	**0.004**	**0.007**	**0.002**	**0.001**
	ae										0.157	0.343	**0.063**	**0.032**
	sg											0.857	0.373	0.183
	a												0.492	0.238
	c													0.297

F-test		nh	nr	na	nbe	nb	ncd	nc	h	ae	sg	a	c	be
	nae	0.292	0.867	0.984	0.291	0.558	**0.026**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
	nh		0.223	0.302	0.998	0.639	0.237	**0.009**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
	nr			0.852	0.222	0.452	**0.017**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
	na				0.300	0.572	**0.028**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
	nbe					0.637	0.238	**0.009**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
	nb						**0.100**	**0.002**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
	ncd							0.145	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
	nc								**0.033**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
	h									**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
	ae										0.765	0.284	0.616	0.455
	sg											0.439	0.839	0.654
	a												0.568	0.744
	c													0.806

Table E3b. Statistical comparison of input features for NN1 networks with similar architecture

Architecture	3 hidden layer nodes, 4 outputs																	
Features	nru	nae	na	nce	nbe	ncde	nsg	nh	ncd	nc	ncfg	h	cd	cfg	ae	sg	a	c
Network	l4i	l5e	l4b	l5g	l5b	l5i	l7d	l6a	l4f	l4d	l7	l3c	l1g	l7b	l2d	l7f	l1d	l1f
Mean error, mm	**1.3**	2.2	3.2	4.7	5.4	6.1	6.7	6.7	7.7	9.0	13.5	7.2	28.1	38.9	46.2	48.9	50.7	61.3
Stdev error, mm	17.3	16.8	16.5	18.0	19.1	**13.7**	20.0	23.2	22.5	24.8	27.3	38.4	94.2	93.3	106.9	105.1	117.3	112.0

t-test	nae	na	nce	nbe	ncde	nsg	nh	ncd	nc	ncfg	h	cd	cfg	ae	sg	a	c
nru	0.546	0.196	**0.092**	**0.021**	**0.046**	**0.007**	**0.089**	**0.009**	**0.007**	**0.000**	0.238	**0.045**	**0.005**	**0.004**	**0.002**	**0.004**	**0.000**
nae		0.331	**0.099**	**0.032**	**0.029**	**0.001**	**0.098**	**0.002**	**0.005**	**0.000**	0.346	**0.053**	**0.007**	**0.005**	**0.003**	**0.005**	**0.001**
na			0.339	0.175	0.160	**0.035**	0.229	**0.042**	**0.030**	**0.001**	0.439	**0.065**	**0.008**	**0.006**	**0.004**	**0.006**	**0.001**
nce				0.652	0.445	0.188	0.450	0.117	0.101	**0.005**	0.645	**0.082**	**0.012**	**0.007**	**0.004**	**0.006**	**0.001**
nbe					0.725	0.170	0.639	0.132	**0.076**	**0.003**	0.740	**0.092**	**0.013**	**0.008**	**0.005**	**0.007**	**0.001**
ncde						0.749	0.806	0.433	0.307	**0.024**	0.850	0.107	**0.018**	**0.010**	**0.006**	**0.009**	**0.001**
nsg							1.000	0.377	0.180	**0.007**	0.929	0.113	**0.019**	**0.010**	**0.007**	**0.010**	**0.001**
nh								0.627	0.446	**0.018**	0.934	0.122	**0.022**	**0.011**	**0.007**	**0.010**	**0.001**
ncd									0.446	**0.006**	0.928	0.126	**0.022**	**0.011**	**0.008**	**0.010**	**0.001**
nc										**0.048**	0.762	0.159	**0.030**	**0.016**	**0.011**	**0.015**	**0.002**
ncfg											0.292	0.261	**0.053**	**0.027**	**0.018**	**0.024**	**0.003**
h												**0.093**	**0.012**	**0.007**	**0.004**	**0.008**	**0.001**
cd													0.138	**0.096**	**0.059**	**0.059**	**0.012**
cfg														0.447	0.241	0.341	**0.047**
ae															0.681	0.569	0.132
sg																0.864	0.172
a																	0.224

Table E3c. Statistical comparison of input features for NN1 networks with similar architecture

Architecture	3 hidden layer nodes, 4 outputs																
Features	nru	nae	na	nce	nbe	ncde	nsg	nh	ncd	nc	ncfg	h	cd	cfg	ae	sg	a	c
Network	l4i	l5e	l4b	l5g	l5b	l5i	l7d	l6a	l4f	l4d	l7	l3c	l1g	l7b	l2d	l7f	l1d	l1f
Mean error, mm	**1.3**	2.2	3.2	4.7	5.4	6.1	6.7	6.7	7.7	9.0	13.5	7.2	28.1	38.9	46.2	48.9	50.7	61.3
Stdev error, mm	17.3	16.8	16.5	18.0	19.1	**13.7**	20.0	23.2	22.5	24.8	27.3	38.4	94.2	93.3	106.9	105.1	117.3	112.0

F-test		nae	na	nce	nbe	ncde	nsg	nh	ncd	nc	ncfg	h	cd	cfg	ae	sg	a	c
	nru	0.853	0.736	0.769	0.484	0.112	0.315	**0.042**	**0.067**	**0.013**	**0.002**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
	nae		0.880	0.632	0.376	0.160	0.234	**0.027**	**0.044**	**0.007**	**0.001**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
	na			0.528	0.300	0.209	0.180	**0.018**	**0.031**	**0.005**	**0.001**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
	nce				0.684	**0.060**	0.476	**0.081**	0.123	**0.027**	**0.004**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
	nbe					**0.023**	0.759	0.179	0.254	**0.070**	**0.014**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
	ncde						**0.010**	**0.000**	**0.001**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
	nsg							0.299	0.404	0.131	**0.031**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
	nh								0.839	0.635	0.260	**0.001**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
	ncd									0.498	0.184	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
	nc										0.514	**0.003**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
	ncfg											**0.018**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
	h												**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
	cd													0.949	0.378	0.446	0.127	0.228
	cfg														0.345	0.409	0.112	0.204
	ae															0.905	0.516	0.743
	sg																0.442	0.655
	a																	0.747

Table E3d. Statistical comparison of input features for NN1 networks with similar architecture

Architecture	4 hidden layer nodes, 4 outputs										
Features	ncde	nce	nh	nbe	nsg	ncd	ncfg	nae	h	cfg	cd
Network	l5j	l5h	l6b	l5c	l7e	l4g	l7a	l5f	l3d	l7c	l1h
Mean error, mm	**2.4**	3.6	4.1	4.5	6.8	7.5	14.5	12.4	11.9	34.8	35.2
Stdev error, mm	**15.9**	18.1	16.4	16.6	19.8	19.4	28.1	41.8	53.6	90.9	99.5

t-test		nce	nh	nbe	nsg	ncd	ncfg	nae	h	cfg	cd
	ncde	0.554	0.344	0.244	**0.024**	**0.003**	**0.000**	**0.091**	0.218	**0.013**	**0.025**
	nce		0.783	0.406	**0.039**	**0.083**	**0.001**	**0.092**	0.258	**0.015**	**0.028**
	nh			0.814	0.181	**0.096**	**0.002**	0.156	0.300	**0.019**	**0.033**
	nbe				**0.069**	**0.079**	**0.000**	0.159	0.330	**0.020**	**0.033**
	nsg					0.654	**0.006**	0.321	0.501	**0.034**	**0.049**
	ncd						**0.007**	0.416	0.586	**0.042**	**0.055**
	ncfg							0.738	0.745	0.104	0.135
	nae								0.921	**0.057**	**0.084**
	h									**0.028**	**0.055**
	cfg										0.975

F-test		nce	nh	nbe	nsg	ncd	ncfg	nae	h	cfg	cd
	ncde	0.381	0.837	0.792	0.137	0.178	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
	nce		0.502	0.540	0.537	0.636	**0.003**	**0.000**	**0.000**	**0.000**	**0.000**
	nh			0.954	0.199	0.254	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
	nbe				0.220	0.278	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
	nsg					0.885	**0.016**	**0.000**	**0.000**	**0.000**	**0.000**
	ncd						**0.011**	**0.000**	**0.000**	**0.000**	**0.000**
	ncfg							**0.006**	**0.000**	**0.000**	**0.000**
	nae								**0.087**	**0.000**	**0.000**
	h									**0.000**	**0.000**
	cfg										0.527

Table E3e. Statistical comparison of input features for NN1 networks with similar architecture

Architecture	2 hidden layers with 3 PE's each, 4 outputs							
Features	nae	nce	nc	naevu	ncde	ncd	cd	c
Network	l5e2	l5g2	l4d2	l5k2	l5i2	l4f2	l1g2	l1f2
Mean error, mm	**5.4**	5.5	5.4	6.3	7.9	10.3	29.8	52.5
Stdev error, mm	15.1	18.5	19.2	**14.6**	18.4	21.9	97.1	119.9

Architecture	2 hidden layers, 4 PE's each, 4 outputs				
Features	ncde	nae	nce	ncd	cd
Network	l5j2	l5f2	l5h2	l4g2	l1h2
Mean error, mm	**2.1**	5.9	8.6	8.4	42.0
Stdev error, mm	18.3	**14.7**	15.3	19.1	108.5

t-test

	nce	nc	naevu	ncde	ncd	cd	c
nae	0.922	0.989	**0.073**	**0.080**	**0.058**	**0.080**	**0.007**
nce		0.953	0.556	**0.050**	**0.079**	**0.079**	**0.006**
nc			0.486	0.131	**0.005**	**0.079**	**0.006**
naevu				0.247	0.105	**0.092**	**0.008**
ncde					0.358	0.113	**0.009**
ncd						0.156	**0.013**
cd							**0.063**

	nae	nce	ncd	cd
ncde	**0.020**	**0.006**	**0.011**	**0.012**
nae		0.259	0.227	**0.024**
nce			0.939	**0.028**
ncd				**0.026**

F-test

	nce	nc	naevu	ncde	ncd	cd	c
nae	0.157	**0.095**	0.824	0.171	**0.011**	**0.000**	**0.000**
nce		0.797	0.102	0.962	0.253	**0.000**	**0.000**
nc			**0.059**	0.761	0.375	**0.000**	**0.000**
naevu				0.112	**0.006**	**0.000**	**0.000**
ncde					0.234	**0.000**	**0.000**
ncd						**0.000**	**0.000**
cd							0.143

	nae	nce	ncd	cd
ncde	0.129	0.213	0.757	**0.000**
nae		0.784	**0.068**	**0.000**
nce			0.121	**0.000**
ncd				**0.000**

Table E4a. Statistical comparison of NN1 network architectures for treatments with common input features.

Features	nae					nh			ncde				na	
Architecture	2	3	4	33	44	2	3	4	3	4	33	44	2	3
Network	L5d	L5e	L5f	L5e2	L5f2	l6	l6a	l6b	l5i	l5j	l5i2	l5j2	l4	l4b
Mean error, mm	2.3	**2.2**	12.4	5.4	5.9	**2.8**	6.7	4.1	6.1	2.4	7.9	**2.1**	3.5	**3.2**
Stdev error, mm	16.6	16.8	41.8	15.1	**14.7**	19.3	23.2	**16.4**	**13.7**	15.9	18.4	18.3	16.7	**16.5**
t-test		3	4	33	44		3	4		4	33	44		
	2	0.891	**0.071**	**0.001**	**0.000**	2	0.192	0.570	3	**0.003**	0.336	**0.065**		0.785
	3		**0.069**	**0.004**	**0.001**	3		0.152	4		**0.014**	0.880		
	4			0.205	0.243				33			**0.001**		
	33				0.192									
F-test		3	4	33	44		3	4		4	33	44		
	2	0.932	**0.000**	0.508	0.391	2	0.208	0.255	3	0.302	**0.043**	**0.048**		0.932
	3		**0.000**	0.455	0.345	3		**0.017**	4		0.318	0.340		
	4			**0.000**	**0.000**				33			0.963		
	33				0.844									

Architecture: 2) 2 H1 PE's 3) 3 H1 PE's 4) 4 H1 PE's 33) 3 H1, 3 H2 PE's 44) 4 H1, 4 H2 PE's

Table E4b. Statistical comparison of NN1 network architectures for treatments with common input features.

Features	nce				nbe			nsg		ncd				
Architecture	3	4	33	44	2	3	4	3	4	2	3	4	33	44
Network	l5g	l5h	l5g2	l5h2	l5a	l5b	l5c	l7d	l7e	l4e	l4f	l4g	l4f2	l4g2
Mean error, mm	4.7	**3.6**	5.5	8.6	4.5	5.4	**4.5**	**6.7**	6.8	**7.4**	7.7	7.5	10.3	8.4
Stdev error, mm	18.0	18.1	18.5	**15.3**	19.4	19.1	**16.6**	20.0	**19.8**	22.9	22.5	19.4	21.9	**19.1**
t-test		4	33	44		3	4				3	4	33	44
	3	**0.004**	0.291	0.125	2	0.383	0.961		0.920	2	0.660	0.935	**0.040**	0.509
	4		**0.017**	**0.049**	3		0.435			3		0.901	0.101	0.638
	33			0.224						4			**0.069**	0.540
										33				**0.044**
F-test		4	33	44		3	4				3	4	33	44
	3	0.983	0.847	0.251	2	0.932	0.279		0.940	2	0.900	0.239	0.737	0.207
	4		0.864	0.243	3		0.319			3		0.292	0.833	0.255
	33			0.181						4			0.399	0.931
										33				0.353

Architecture: 2) 2 H1 PE's 3) 3 H1 PE's 4) 4 H1 PE's 33) 3 H1, 3 H2 PE's 44) 4 H1, 4 H2 PE's

Table E4c. Statistical comparison of NN1 network architectures for treatments with common input features.

Features	nc			h			ncfg		cd				cfg	
Architecture	2	3	33	2	3	4	3	4	3	4	33	44	3	4
Network	l4c	l4d	l4d2	l3b	l3c	l3d	l7	l7a	l1g	l1h	l1g2	l1h2	l7b	l7c
Mean error, mm	10.7	9.0	**5.4**	**7.1**	7.2	11.9	**13.5**	14.5	**28.1**	35.2	29.8	42.0	38.9	**34.8**
Stdev error, mm	28.3	24.8	**19.2**	38.5	**38.4**	53.6	**27.3**	28.1	**94.2**	99.5	97.1	108.5	93.3	**90.9**

t-test		3	33		3	4			4	33	44	
	2	0.509	**0.055**	2	0.775	0.382	**0.000**	3	0.230	0.649	0.133	0.725
	3		**0.038**	3		0.388		4		0.308	0.539	
								33			0.241	

F-test		3	33		3	4			4	33	44	
	2	0.262	**0.002**	2	0.986	**0.023**	0.844	3	0.700	0.832	0.325	0.853
	3		**0.077**	3		**0.022**		4		0.863	0.548	
								33			0.440	

Architecture: 2) 2 H1 PE's 3) 3 H1 PE's 4) 4 H1 PE's 33) 3 H1, 3 H2 PE's 44) 4 H1, 4 H2 PE's

Table E4d. Statistical comparison of NN1 network architectures for treatments with common input features.

Features	ae		sg		a		c		
Architecture	2	3	2	3	2	3	2	3	33
Network	l2c	l2d	l7f	l7g	l1c	l1d	l1e	l1f	l1f2
Mean error, mm	**40.9**	46.2	**48.9**	49.0	50.9	**50.7**	55.7	61.3	**52.5**
Stdev error, mm	**100.6**	106.9	105.1	**105.0**	117.4	**117.3**	**108.1**	112.0	119.9
t-test								3	33
		0.452		**0.032**		0.307	2	0.287	0.739
							3		0.420
F-test								3	33
		0.674		0.998		0.998	2	0.805	0.472
							3		0.636

Architecture: 2) 2 H1 PE's 3) 3 H1 PE's 33) 3 H1, 3 H2 PE's

Table E5a. Comparison of NN2 features for treatments with common architecture and NN1 features.

NN1 features	na				na			na	
NN2 architecture	5 hidden layer PE's				3 hidden layer PE's			15 hidden PE's	
Network	h4h	h4c	h4f	h4a	h4g	h4e	h4d	h4b	h4
NN2 features	C	CF	CFSR	CFR	CFSR	CF	CFR	CF	CFR
Mean error, mm	-0.2	**0.2**	0.2	2.4	-0.3	**0.1**	1.0	**0.4**	0.6
Stdev error, mm	18.0	18.0	18.6	**15.8**	17.6	**17.3**	18.1	18.0	**17.9**

t-test		CF	CFSR	CFR		CF	CFR	
	C	0.785	0.601	**0.025**	CFSR	0.271	0.151	0.842
	CF		0.939	0.109	CF		0.258	
	CFSR			**0.030**				

F-test		CF	CFSR	CFR		CF	CFR	
	C	0.996	0.808	0.370	CFSR	0.921	0.844	0.955
	CF		0.812	0.368	CF		0.767	
	CFSR			0.255				

Table E5b. Comparison of NN2 features for treatments with common architecture and NN1 features.

NN1 features	nh			nh			nae	
NN2 architecture	5 hidden layer PE's			3 hidden layer PE's			3 hidden PE's	
Network	h6a	h6	h6b	h6c	h6d	h6e	h5b	h5c
NN2 features	CF	CFR	C	CFR	CF	C	CFR	CF
Mean error, mm	**4.7**	5.6	7.2	**3.3**	5.1	10.0	**0.0**	4.0
Stdev error, mm	**24.9**	25.8	24.9	**25.4**	27.6	26.4	17.7	**16.4**
t-test		CFR	C		CF	C		
	CF	0.231	**0.000**	CFR	**0.047**	**0.000**		**0.004**
	CFR		**0.004**	CF		**0.000**		
		CFR	C		CF	C		
F-test	CF	0.801	0.998	CFR	0.562	0.798		0.607
	CFR		0.803	CF		0.746		

Table E6. Statistical comparison of NN2 networks with inputs from different NN1 networks.

NN2 features	CFR						CF					
NN2 architecture	5 hidden layer PE's			3 hidden layer PE's			5 hidden layer PE's			3 hidden layer PE's		
Net	h4a	h6	h5	h5b	h4d	h6c	h4c	h5a	h6a	h4e	h5c	h6d
NN1 features	na	nh	nbe	nae	na	nh	na	nae	nh	na	nae	nh
Mean error, mm	**2.4**	5.6	9.2	**0.0**	1.0	3.3	**0.2**	3.0	4.7	**0.1**	4.0	5.1
Stdev error, mm	**15.8**	25.8	31.0	**17.7**	18.1	25.4	18.0	**16.8**	24.9	17.3	**16.4**	27.6
t-test		nh	nbe		na	nh		nae	nh		nae	nh
	na	0.193	**0.061**	nae	0.353	0.225	na	**0.016**	0.109	na	**0.000**	**0.099**
	nh		0.201	na		0.401	nae		0.481	nae		0.719
F-test		nh	nbe		na	nh		nae	nh		nae	nh
	na	**0.001**	**0.000**	nae	0.875	**0.012**	na	0.644	**0.025**	na	0.707	**0.001**
	nh		0.203	na		**0.019**	nae		**0.007**	nae		**0.000**

Table E7a. Statistical comparison of NN2 network architecture for common NN1 and NN2 input features.

NN1 features	na										
NN2 features	CFR						CF			CFSR	
Network	h4i	h4k	h4	h4j	h4d	h4a	h4e	h4c	h4b	h4f	h4g
Connectivity	line	line	full	line	full	full	full	full	full	full	full
Hidden PE's	5	5	15	5	3	5	3	5	15	5	3
In/hid connections	105	45	675	60	135	225	90	150	450	300	180
Mean error, mm	**0.1**	0.1	0.6	0.9	1.0	2.4	**0.1**	0.2	0.4	**0.2**	-0.3
Stdev error, mm	18.1	18.5	17.9	18.0	18.1	**15.8**	**17.3**	18.0	18.0	18.6	**17.6**

t-test		45	675	60	135	225		150	450	
	105	0.948	0.600	0.321	0.306	0.106	90	0.817	0.609	0.575
	45		0.554	0.418	0.365	**0.022**	150		0.738	
	675			0.745	0.504	**0.072**				
	60				0.958	0.233				
	135					0.248				

F-test		45	675	60	135	225		150	450	
	105	0.884	0.918	0.954	0.988	0.338	90	0.800	0.792	0.695
	45		0.804	0.839	0.873	0.270	150		0.991	
	675			0.964	0.930	0.392				
	60				0.966	0.367				
	135					0.345				

Table E7b. Statistical comparison of NN2 network architecture for common NN1 and NN2 input features.

NN1 features	nae		nh					
NN2 features	CF		CFR		CF		C	
Network	h5a	h5c	h6	h6c	h6a	h6d	h6b	h6e
Connectivity	full	full	full	full	full	full	full	full
Hidden PE's	5	3	5	3	5	3	5	3
In/hid connections	150	90	225	135	150	90	75	45
Mean error, mm	**3.0**	4.0	5.6	**3.3**	**4.7**	5.1	**7.2**	10.0
Stdev error, mm	16.8	**16.4**	25.8	**25.4**	**24.9**	27.6	**24.9**	26.4
t-test	0.338		**0.053**		0.754		**0.013**	
F-test	0.868		0.916		0.468		0.689	

Table E8. Statistical comparison of NN2 networks with NN1 network providing input to NN2.

NN1/NN2	NN1	NN2											
Network	l4	h4g	h4h	h4e	h4i	h4k	h4c	h4f	h4b	h4	h4j	h4d	h4a
Mean error, mm	17.9	-0.3	-0.2	**0.1**	0.1	0.1	0.2	0.2	0.4	0.6	0.9	1.0	2.4
Stdev error, mm	66.1	17.6	18.0	17.3	18.1	18.5	18.0	18.6	18.0	17.9	18.0	18.1	**15.8**
t-test		**0.055**	**0.041**	**0.057**	**0.062**	**0.043**	**0.062**	**0.044**	**0.061**	**0.052**	**0.074**	**0.058**	**0.082**
F-test		**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
Network	l1a	h1a	h1										
Mean error, mm	61.7	**3.1**	5.7										
Stdev error, mm	121.5	**15.4**	15.5										
t-test		**0.002**	**0.003**										
F-test		**0.000**	**0.000**										
Network	l5f	h5b	h5a	h5c									
Mean error, mm	9.0	**0.0**	3.0	4.0									
Stdev error, mm	42.6	17.7	16.8	**16.4**									
t-test		0.112	0.293	0.368									
F-test		**0.000**	**0.000**	**0.000**									
Network	l6a	h6c	h6a	h6d	h6	h6b	h6e						
Mean error, mm	6.3	**3.3**	4.7	5.1	5.6	7.2	10.0						
Stdev error, mm	**22.8**	25.4	24.9	27.6	25.8	24.9	26.4						
t-test		**0.072**	0.231	0.554	0.601	0.450	**0.028**						
F-test		0.457	0.550	0.187	0.396	0.549	0.318						

Table E9a. Statistical comparison of wide FOV NN3 features for treatments with common architecture.

FOV, mm	140																	
Architecture	5 hidden layer PE's							10 hidden layer PE's							15 hidden layer PE's			
Network	w1	w16	w4	w5	w17	w2	w3	w16a	w5a	w17a	w2a	w4a	w1a	w3a	w1b	w5b	w3b	w2b
Features	a0	a2d0	m2	a2	m2d0	a5	m5	a2d0	a2	m2d0	a5	m2	a0	m5	a0	a2	m5	a5
Mean error, mm	-2.7	**1.7**	2.3	2.9	3.6	3.9	4.2	**2.4**	2.8	2.8	3.0	3.6	3.9	4.6	-3.0	**2.6**	3.5	3.8
Stdev error, mm	28.6	13.1	14.6	15.3	13.3	12.8	**12.5**	14.2	**12.9**	13.5	18.2	13.8	28.2	13.9	36.0	**12.3**	13.2	16.6

t-test

	a2d0	m2	a2	m2d0	a5	m5
a0	0.224	0.271	0.111	0.121	**0.076**	**0.093**
a2d0		0.801	0.329	0.221	**0.005**	0.109
m2			0.784	0.486	0.455	0.278
a2				0.668	0.463	0.397
m2d0					0.800	0.435
a5						0.874

	a2	m2d0	a5	m2	a0	m5
a2d0	0.629	0.684	0.864	0.509	0.725	0.168
a2		1.000	0.956	0.647	0.789	0.249
m2d0			0.956	0.696	0.801	0.193
a5				0.837	0.832	0.630
m2					0.947	0.643
a0						0.872

	a2	m5	a5
a0	0.297	0.244	0.233
a2		0.428	0.511
m5			0.856

F-test

	a2d0	m2	a2	m2d0	a5	m5
a0	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
a2d0		0.499	0.328	0.907	0.889	0.786
m2			0.761	0.576	0.415	0.344
a2				0.389	0.264	0.212
m2d0					0.798	0.697
a5						0.894

	a2	m2d0	a5	m2	a0	m5
a2d0	0.549	0.733	0.127	0.866	**0.000**	0.880
a2		0.795	0.035	0.666	**0.000**	0.653
m2d0			0.063	0.864	**0.000**	0.849
a5				**0.091**	**0.008**	**0.094**
m2					**0.000**	0.985
a0						**0.000**

	a2	m5	a5
a0	**0.000**	**0.000**	**0.000**
a2		0.644	**0.065**
m5			0.164

Table E9b. Statistical comparison of wide FOV NN3 features for treatments with common architecture.

FOV, mm	140														
Architecture	5 hidden layer 1 PE's						10 hidden layer 1 PE's						15 H1 PE's		
	3 hidden layer 2 PE's						5 hidden layer 2 PE's						7 H2 PE's		
Network	w17l	w5l	w2l	w4l	w16l	w3l	w17m	w2m	w16m	w5m	w4m	w3m	w2n	w5n	w3n
Features	m2d0	a2	a5	m2	a2d0	m5	m2d0	a5	a2d0	a2	m2	m5	a5	a2	m5
Mean error, mm	**0.6**	1.0	1.1	1.2	1.7	2.9	**0.8**	1.1	1.7	3.1	3.6	6.2	**0.5**	2.1	7.2
Stdev error, mm	13.1	**9.8**	18.2	15.3	15.9	15.3	14.4	17.3	13.5	14.5	14.9	**12.8**	16.1	**14.1**	16.0
t-test		a2	a5	m2	a2d0	m5		a5	a2d0	a2	m2	m5		a2	m5
	m2d0	0.838	0.862	0.699	0.449	0.247	m2d0	0.915	0.692	0.317	0.277	**0.004**	a5	0.535	**0.007**
	a2		0.963	0.925	0.739	0.369	a5		0.793	0.465	0.361	**0.029**	a2		**0.025**
	a5			0.976	0.814	0.536	a2d0			0.357	0.405	**0.013**			
	m2				0.796	0.177	a2				0.863	0.123			
	a2d0					0.592	m2					0.178			
		a2	a5	m2	a2d0	m5		a5	a2d0	a2	m2	m5		a2	m5
F-test	m2d0	**0.078**	**0.044**	0.329	0.236	0.326	m2d0	0.250	0.673	0.971	0.846	0.479	a5	0.407	0.972
	a2		**0.000**	**0.007**	**0.004**	**0.007**	a5		0.117	0.265	0.338	**0.064**	a2		0.427
	a5			0.294	0.401	0.296	a2d0			0.647	0.538	0.775			
	m2				0.833	0.996	a2				0.874	0.457			
	a2d0					0.837	m2					0.368			

Table E10a. Statistical comparison of wide FOV NN3 network architecture for treatments with common inputs.

FOV, mm	140														
Features	a5						m5						a0		
Network	w2n	w2m	w2l	w2a	w2b	w2	w3l	w3b	w3	w3a	w3m	w3n	w1b	w1	w1a
Architecture	F	E	D	B	C	A	D	C	A	B	E	F	C	A	B
Mean error, mm	**0.5**	1.1	1.1	3.0	3.8	3.9	**2.9**	3.5	4.2	4.6	6.2	7.2	-3.0	**-2.7**	3.9
Stdev error, mm	16.1	17.3	18.2	18.2	16.6	**12.8**	15.3	13.2	12.5	13.9	**12.8**	16.0	36.0	**28.6**	28.2
t-test		E	D	B	C	A		C	A	B	E	F		A	B
	F	0.640	0.736	0.346	0.185	**0.043**	D	0.741	0.468	0.419	**0.079**	**0.078**	C	0.962	0.267
	E		0.989	0.512	0.277	0.111	C		0.336	0.325	**0.009**	**0.031**	A		0.095
	D			0.517	0.347	0.237	A			0.695	**0.036**	**0.095**			
	B				0.813	0.734	B				0.309	**0.053**			
	C					0.952	E					0.596			
F-test		E	D	B	C	A		C	A	B	E	F		A	B
	F	0.652	0.460	0.451	0.864	0.154	D	0.360	0.211	0.536	0.271	0.785	C	0.154	0.128
	E		0.773	0.762	0.780	**0.062**	C		0.734	0.767	0.851	0.236	A		0.923
	D			0.988	0.571	**0.032**	A			0.525	0.879	0.128			
	B				0.560	**0.030**	B				0.628	0.373			
	C					0.111	E					0.170			

Architecture: A) 5 H1 PE's B) 10 H1 PE's C) 15 H1 PE's D) 5 H1, 3 H2 PE's E) 10 H1, 5 H2 PE's F) 15 H1, 7 H2 PE's

Table E10b. Statistical comparison of wide FOV NN3 network architecture for treatments with common inputs.

Features	a2															
FOV, mm	140						70						210			
Network	w5l	w5n	w5b	w5a	w5	w5m	w7b	w7m	w7	w7n	w7a	w7l	w8	w8a	w8l	w8m
Architecture	D	F	C	B	A	E	C	E	A	F	B	D	A	B	D	E
Mean error, mm	**1.0**	2.1	2.6	2.8	2.9	3.1	-0.4	**0.2**	0.8	0.8	1.1	3.5	-0.7	**-0.2**	1.0	1.3
Stdev error, mm	**9.8**	14.1	12.3	12.9	15.3	14.5	19.4	21.7	16.9	21.5	15.1	**13.3**	14.3	**13.9**	14.3	15.0

t-test

	F	C	B	A	E
D	0.590	0.336	0.292	0.377	0.168
F		0.572	0.379	0.652	0.607
C			0.706	0.850	0.782
B				0.947	0.850
A					0.894

	E	A	F	B	D
C	0.873	0.416	0.652	0.346	0.091
E		0.875	0.889	0.774	0.281
A			0.993	0.769	0.158
F				0.913	0.436
B					0.184

	B	D	E
A	0.828	0.516	0.412
B		0.082	0.426
D			0.864

	B	D	E
A	0.859	0.984	0.757
B		0.843	0.626
D			0.772

F-test

	F	C	B	A	E
D	**0.027**	0.170	**0.094**	**0.007**	**0.018**
F		0.392	0.584	0.604	0.871
C			0.757	0.171	0.309
B				0.287	0.478
A					0.722

	E	A	F	B	D
C	0.482	0.404	0.528	0.126	**0.021**
E		0.126	0.943	**0.027**	**0.003**
A			0.144	0.484	0.137
F				**0.032**	**0.004**
B					0.428

Architecture: A) 5 H1 PE's B) 10 H1 PE's C) 15 H1 PE's D) 5 H1, 3 H2 PE's E) 10 H1, 5 H2 PE's F) 15 H1, 7 H2 PE's

Table E10c. Statistical comparison of wide FOV NN3 network architecture for treatments with common inputs.

Features	a2											
FOV, mm	280				100				80			
Network	w10l	w10a	w10m	w10	w12	w12l	w12a	w12m	w14	w14a	w14l	w14m
Architecture	D	B	E	A	A	D	B	E	A	B	D	E
Mean error, mm	**1.8**	2.4	3.3	3.5	**2.2**	2.3	3.3	3.9	-0.3	-0.2	**0.2**	1.6
Stdev error, mm	14.3	14.4	14.1	**13.5**	**12.8**	15.4	15.2	12.9	**12.6**	14.5	13.9	14.2
t-test		B	E	A		D	B	E		B	D	E
	D	0.348	0.149	**0.005**	A	0.947	0.421	**0.014**	A	0.951	0.750	0.297
	B		0.348	**0.026**	D		0.550	0.269	B		0.600	0.402
	E			0.791	B			0.558	D			0.486
F-test		B	E	A		D	B	E		B	D	E
	D	0.949	0.924	0.716	A	0.254	0.291	0.973	A	0.364	0.537	0.452
	B		0.874	0.669	D		0.932	0.269	B		0.770	0.876
	E			0.788	B			0.307	D			0.892

Architecture: A) 5 H1 PE's B) 10 H1 PE's D) 5 H1, 3 H2 PE's E) 10 H1, 5 H2 PE's

Table E10d. Statistical comparison of wide FOV NN3 network architecture for treatments with common inputs.

Features	m2															
FOV, mm	140				70				210				280			
Network	w4l	w4	w4a	w4m	w6m	w6l	w6a	w6	w9	w9l	w9m	w9a	w11a	w11	w11m	w11l
Architecture	D	A	B	E	E	D	B	A	A	D	E	B	B	A	E	D
Mean error, mm	**1.2**	2.3	3.6	3.6	**0.2**	1.2	1.5	1.8	**0.1**	0.5	1.1	1.6	**3.5**	4.2	4.4	4.6
Stdev error, mm	15.3	14.6	**13.8**	14.9	18.8	16.9	15.5	**14.8**	**14.9**	15.8	17.1	15.1	13.4	**11.6**	12.2	12.5
t-test		A	B	E		D	B	A		D	E	B		A	E	D
	D	0.311	0.213	0.270	E	0.623	0.386	0.358	A	0.816	0.610	0.496	B	0.464	0.373	0.311
	A		0.412	0.512	D		0.822	0.648	D		0.787	0.660	A		0.890	0.658
	B			0.981	B			0.744	E			0.766	E			0.743
F-test		A	B	E		D	B	A		D	E	B		A	E	D
	D	0.763	0.527	0.844	E	0.513	0.235	0.138	A	0.726	0.401	0.933	B	0.362	0.568	0.677
	A		0.741	0.916	D		0.592	0.405	D		0.623	0.790	A		0.732	0.619
	B			0.662	B			0.766	E			0.449	E			0.877

Architecture: A) 5 H1 PE's B) 10 H1 PE's D) 5 H1, 3 H2 PE's E) 10 H1, 5 H2 PE's

Table E10e. Statistical comparison of wide FOV NN3 network architecture for treatments with common inputs.

Features	m2								a2d0				m2d0			
FOV, mm	100				80				140				140			
Network	w13m	w13l	w13a	w13	w15l	w15a	w15	w15m	w16	w16m	w16l	w16a	w17l	w17m	w17a	w17
Architecture	E	D	B	A	D	B	A	E	A	E	D	B	D	E	B	A
Mean error, mm	**1.5**	3.9	4.5	4.6	**0.3**	0.7	0.8	6.4	**1.7**	1.7	1.7	2.4	**0.6**	0.8	2.8	3.6
Stdev error, mm	16.7	16.8	**11.9**	12.3	12.8	12.7	**10.4**	12.7	**13.1**	13.5	15.9	14.2	**13.1**	14.4	13.5	13.3
t-test		D	B	A		B	A	E		E	D	B		E	B	A
	E	**0.062**	**0.084**	**0.084**	D	0.801	0.722	**0.008**	A	0.985	1.000	0.687	D	0.883	0.229	**0.065**
	D		0.741	0.708	B		0.889	**0.005**	E		0.989	0.713	E		0.333	0.204
	B			0.830	A			**0.014**	D			0.693	B			0.533
F-test		D	B	A		B	A	E		E	D	B		E	B	A
	E	0.958	**0.040**	**0.059**	D	0.965	0.190	0.964	A	0.868	0.236	0.607	D	0.556	0.862	0.907
	D		**0.036**	**0.052**	B		0.205	0.999	E		0.307	0.728	E		0.678	0.637
	B			0.868	A			0.205	D			0.500	B			0.955

Architecture: A) 5 H1 PE's B) 10 H1 PE's D) 5 H1, 3 H2 PE's E) 10 H1, 5 H2 PE's

Table E11. Statistical comparison of wide FOV NN3 network lines input.

Features	a2							
Architecture	A		B		C		D	
Network	w12	w5	w12a	w5a	w12l	w5l	w12m	w5m
Lines	21	15	21	15	21	15	21	15
FOV, mm	100	140	100	140	100	140	100	140
Mean error, mm	**2.2**	2.9	3.3	**2.8**	2.3	**1.0**	3.9	**3.1**
Stdev error, mm	**12.8**	15.3	15.2	**12.9**	15.4	**9.8**	**12.9**	14.5
t-test		0.605		0.690		0.578		0.698
F-test		0.262		0.318		**0.006**		0.464

Features	m2							
Architecture	A		B		C		D	
Network	w13	w4	w13a	w4a	w13l	w4l	w13m	w4m
Lines	21	15	21	15	21	15	21	15
FOV, mm	100	140	100	140	100	140	100	140
Mean error, mm	4.6	**2.3**	4.5	**3.6**	3.9	**1.2**	**1.5**	3.6
Stdev error, mm	**12.3**	14.6	**11.9**	13.8	16.8	**15.3**	16.7	**14.9**
t-test		0.160		0.539		0.278		0.405
F-test		0.280		0.359		0.569		0.476

Features	a2							
Architecture	A		B		C		D	
Network	w7	w14	w7a	w14a	w7l	w14l	w7m	w14m
Lines	15	9	15	9	15	9	15	9
FOV, mm	70	80	70	80	70	80	70	80
Mean error, mm	0.8	**-0.3**	1.1	**-0.2**	3.5	**0.2**	**0.2**	1.6
Stdev error, mm	16.9	**12.6**	15.1	**14.5**	**13.3**	13.9	21.7	**14.2**
t-test		0.522		**0.061**		**0.068**		0.636
F-test		**0.065**		0.805		0.800		**0.009**

Features	m2							
Architecture	A		B		C		D	
Network	w6	w15	w6a	w15a	w6l	w15l	w6m	w15m
Lines	15	9	15	9	15	9	15	9
FOV, mm	70	80	70	80	70	80	70	80
Mean error, mm	1.8	**0.8**	1.5	**0.7**	1.2	**0.3**	**0.2**	6.4
Stdev error, mm	14.8	**10.4**	15.5	**12.7**	16.9	**12.8**	18.8	**12.7**
t-test		0.630		0.683		0.686		**0.011**
F-test		**0.028**		0.216		**0.085**		**0.016**

Architecture: A) 5 H1 PE's B) 10 H1 PE's
D) 5 H1, 3 H2 PE's E) 10 H1, 5 H2 PE's

Table E12a. Statisitcal comparison of wide FOV NN3 network field of view.

Features	a2															
Architecture	5 hidden PE's				10 hidden PE's				5 H1, 3 H2 PE's				10 H1, 5 H2 PE's			
Network	w8	w7	w5	w10	w8a	w7a	w10a	w5a	w5l	w8l	w10l	w7l	w7m	w8m	w5m	w10m
FOV, mm	210	70	140	280	210	70	280	140	140	210	280	70	70	210	140	280
Mean error, mm	**-0.7**	0.8	2.9	3.5	**-0.2**	1.1	2.4	2.8	**1.0**	1.0	1.8	3.5	**0.2**	1.3	3.1	3.3
Stdev error, mm	14.3	16.9	15.3	**13.5**	13.9	15.1	14.4	**12.9**	**9.8**	14.3	14.3	13.3	21.7	15.0	14.5	**14.1**
t-test		70	140	280		70	280	140		210	280	70		210	140	280
	210	0.531	0.144	**0.047**	210	0.507	0.210	**0.084**	140	0.989	0.676	0.247	70	0.712	0.414	0.374
	70		0.288	0.199	70		0.571	0.359	210		0.708	0.257	210		0.349	0.313
	140			0.665	280			0.753	280			0.287	140			0.929
F-test		70	140	280		70	280	140		210	280	70		210	140	280
	210	0.297	0.665	0.708	210	0.601	0.817	0.650	140	**0.021**	**0.022**	**0.063**	70	**0.024**	**0.013**	**0.008**
	70		0.540	0.157	70		0.770	0.329	210		0.976	0.640	210		0.817	0.678
	140			0.419	280			0.494	280			0.662	140			0.854

Table E12b. Statisitcal comparison of wide FOV NN3 network field of view.

Features	m2															
Architecture	5 hidden PE's				10 hidden PE's				5 H1, 3 H2 PE's				10 H1, 5 H2 PE's			
Network	w9	w6	w4	w11	w6a	w9a	w11a	w4a	w9l	w4l	w6l	w11l	w6m	w9m	w4m	w11m
FOV, mm	210	70	140	280	70	210	280	140	210	140	70	280	70	210	140	280
Mean error, mm	**0.1**	1.8	2.3	4.2	**1.5**	1.6	3.5	3.6	**0.5**	1.2	1.2	4.6	**0.2**	1.1	3.6	4.4
Stdev error, mm	14.9	14.8	14.6	**11.6**	15.5	15.1	**13.4**	13.8	15.8	15.3	16.9	**12.5**	18.8	17.1	14.9	**12.2**
t-test		70	140	280		210	280	140		140	70	280		210	140	280
	210	0.259	0.188	**0.035**	70	0.969	0.233	0.312	210	0.823	0.623	**0.051**	70	0.630	0.224	**0.085**
	70		0.811	0.151	210		0.164	0.285	140		0.983	**0.074**	210		0.348	0.111
	140			0.286	280			0.925	70			0.109	140			0.674
F-test		70	140	280		210	280	140		140	70	280		210	140	280
	210	0.962	0.896	0.118	70	0.868	0.364	0.477	210	0.858	0.662	0.156	70	0.549	0.144	**0.008**
	70		0.933	0.129	210		0.457	0.586	140		0.538	0.214	210		0.387	**0.040**
	140			0.151	280			0.843	70			0.064	140			0.229

Table E13a. Statistical comparison of narrow FOV NN3 network features.

FOV, mm	15																	
Hidden layer PE's	6	6	6	6	7	7	12	12	12	12	14	14	18	18	18	18	21	21
Network	m1c	m4c	m8c	m7c	m4	m1	m8d	m1d	m4d	m7d	m4a	m1a	m8e	m1e	m4e	m7e	m1b	m4b
Features	bc	cd	ef	ce	acd	abc	ef	bc	cd	ce	acd	abc	ef	bc	cd	ce	abc	acd
Mean error, mm	0.6	0.5	**-0.1**	-0.7	-1.6	-2.2	0.7	0.4	**-0.2**	-0.4	-1.0	-2.0	0.7	0.3	**0.2**	-0.4	-1.3	-2.2
Stdev error, mm	13.4	13.4	**12.6**	19.8	12.7	13.2	12.9	12.6	13.4	19.9	13.1	**12.5**	13.1	13.6	13.4	20.2	**12.9**	13.4

t-test		cd	ef	ce	acd	abc		bc	cd	ce	acd	abc		bc	cd	ce	abc	acd
	bc	0.687	0.621	0.692	**0.089**	**0.034**	ef	0.801	0.566	0.729	0.238	**0.066**	ef	0.810	0.752	0.737	**0.161**	0.050
	cd		0.691	0.727	0.128	0.051	bc		0.555	0.792	0.164	**0.015**	bc		0.528	0.824	**0.210**	0.057
	ef			0.860	0.264	0.125	cd			0.942	0.561	0.185	cd			0.851	**0.257**	0.075
	ce				0.775	0.626	ce				0.861	0.626	ce				0.778	0.562
	acd					0.146	acd					**0.052**	abc					0.101

F-test		cd	ef	ce	acd	abc		bc	cd	ce	acd	abc		bc	cd	ce	abc	acd
	bc	0.985	0.651	**0.007**	0.688	0.924	ef	0.862	0.795	**0.003**	0.934	0.842	ef	0.832	0.895	**0.003**	0.916	0.910
	cd		0.638	**0.008**	0.675	0.909	bc		0.664	**0.002**	0.798	0.980	bc		0.936	**0.006**	0.751	0.921
	ef			**0.002**	0.959	0.721	cd			**0.007**	0.859	0.646	cd			**0.005**	0.812	0.985
	ce				**0.002**	**0.006**	ce				**0.004**	**0.002**	ce				**0.002**	**0.004**
	acd					0.760	acd					0.778	abc					0.827

Table E13b. Statistical comparison of narrow FOV NN3 network features.

FOV, mm	29											
Hidden layer PE's	6	7	7	6	14	12	14	12	21	18	21	18
Network	m2c	m5	m2	m5c	m5a	m2d	m2a	m5d	m5b	m2e	m2b	m5e
Features	bc	acd	abc	cd	acd	bc	abc	cd	acd	bc	abc	cd
Mean error, mm	**-1.3**	-2.0	-2.7	-3.1	1.2	**0.3**	-0.7	-1.9	**-0.2**	-0.6	-1.1	-2.5
Stdev error, mm	13.0	**11.7**	13.1	12.2	12.8	12.8	**10.8**	12.6	12.6	12.8	**10.3**	12.2
t-test		acd	abc	cd		bc	abc	cd		bc	abc	cd
	bc	0.627	0.123	**0.009**	acd	0.154	0.209	**0.000**	acd	0.802	0.495	0.142
	acd		0.628	0.492	bc		0.473	**0.000**	bc		0.669	**0.001**
	abc			0.745	abc			0.387	abc			0.289
F-test		acd	abc	cd		bc	abc	cd		bc	abc	cd
	bc	0.465	0.944	0.689	acd	0.991	0.242	0.930	acd	0.883	0.160	0.852
	acd		0.423	0.741	bc		0.247	0.940	bc		0.121	0.739
	abc			0.638	abc			0.279	abc			0.222

Table E13c. Statistical comparison of narrow FOV NN3 network features.

FOV, mm	43											
Hidden layer PE's	7	7	6	6	14	14	12	12	21	21	18	18
Network	m6	m3	m3c	m6c	m3a	m6a	m3d	m6d	m3b	m6b	m3e	m6e
Features	acd	abc	bc	cd	abc	acd	bc	cd	abc	acd	bc	cd
Mean error, mm	**-2.0**	-2.3	-3.4	-5.1	**-0.6**	-0.7	-2.1	-3.6	**-1.4**	-1.6	-3.0	-3.6
Stdev error, mm	10.7	12.5	12.7	**10.8**	**10.7**	10.7	13.7	10.9	**10.5**	10.6	13.4	10.6
t-test		abc	bc	cd		acd	bc	cd		acd	bc	cd
	acd	0.885	0.438	**0.031**	abc	0.498	0.306	**0.003**	abc	0.255	0.274	**0.027**
	abc		0.257	**0.060**	acd		0.347	**0.005**	acd		0.350	**0.055**
	bc			0.153	bc			0.321	bc			0.697
F-test		abc	bc	cd		acd	bc	cd		acd	bc	cd
	acd	0.288	0.231	0.934	abc	0.996	**0.085**	0.916	abc	0.968	**0.094**	0.941
	abc		0.891	0.327	acd		**0.085**	0.913	acd		0.102	0.974
	bc			0.264	bc			0.106	bc			0.109

Table E14a. Statistical comparison of narrow FOV NN3 network architectures.

Features	bc																		
FOV, mm	15							29						43					
Network	m1e	m1d	m1c	m1i	m1f	m1h	m1g	m2c	m2e	m2g	m2h	m2f	m2d	m3f	m3d	m3g	m3h	m3e	m3c
Architecture	e	d	c	i	f	h	g	c	e	g	h	f	d	f	d	g	h	e	c
Mean error	**0.3**	0.4	0.6	1.5	1.7	2.7	3.9	-1.3	-0.6	-0.1	**0.0**	0.3	0.3	**-1.0**	-2.1	-2.2	-2.7	-3.0	-3.4
Stdev error	13.6	12.6	13.4	13.6	13.4	13.6	**12.3**	13.0	12.8	**12.5**	12.7	12.6	12.8	12.7	13.7	14.2	**12.5**	13.4	12.7
t-test		d	c	i	f	h	g		e	g	h	f	d		d	g	h	e	c
	e	0.935	0.179	**0.001**	**0.000**	**0.000**	**0.000**	c	**0.000**	**0.084**	**0.018**	**0.014**	**0.000**	f	0.192	0.206	**0.000**	**0.018**	**0.000**
	d		0.814	0.300	0.212	**0.027**	**0.000**	e		0.460	0.250	0.146	**0.000**	d		0.942	0.482	**0.000**	**0.031**
	c			**0.041**	**0.007**	**0.000**	**0.000**	g			0.612	**0.000**	0.451	g			0.574	0.148	0.173
	i				0.503	**0.017**	**0.006**	h				0.129	0.517	h				0.688	0.158
	f					**0.008**	**0.009**	f					1.000	e					0.476
	h						0.170												
F-test		d	c	i	f	h	g		e	g	h	f	d		d	g	h	e	c
	e	0.607	0.937	0.979	0.922	0.982	0.487	c	0.956	0.800	0.886	0.831	0.925	f	0.581	0.440	0.939	0.698	0.979
	d		0.664	0.589	0.678	0.592	0.856	e		0.843	0.930	0.874	0.969	d		0.826	0.530	0.870	0.599
	c			0.916	0.984	0.919	0.537	g			0.912	0.968	0.873	g			0.397	0.701	0.456
	i				0.901	0.998	0.471	h				0.944	0.961	h				0.642	0.918
	f					0.903	0.550	f					0.905	e					0.718
	h						0.473												

Architecture: c) 6 H1 PE's d) 12 H1 PE's e) 18 H1 PE's f) 6 H1, 3 H2 PE's g) 12 H1, 4 H2 PE's h) 18 H1, 6 H2 PE's i) 10 H1 PE's

Table E14b. Statistical comparison of narrow FOV NN3 network architectures.

Features	cd																			
FOV, mm	15									29						43					
Network	m4d	m4e	m4c	m4g	m4k	m4i	m4j	m4h	m4f	m5f	m5h	m5g	m5d	m5e	m5c	m6f	m6h	m6g	m6e	m6d	m6c
Architecture	d	e	c	g	k	i	j	h	f	f	h	g	d	e	c	f	h	g	e	d	c
Mean error	-0.2	**0.2**	0.5	0.6	0.7	1.2	1.3	2.1	2.9	**0.1**	-0.1	-0.4	-1.9	-2.5	-3.1	**-1.2**	-2.3	-2.7	-3.6	-3.6	-5.1
Stdev error	13.4	13.4	13.4	12.9	12.6	11.8	**11.4**	13.4	12.4	13.0	12.9	12.7	12.6	**12.2**	12.2	13.1	12.8	12.6	**10.6**	10.9	10.8

t-test		e	c	g	k	i	j	h	f
	d	**0.000**	**0.014**	0.669	0.626	0.318	0.401	**0.000**	**0.001**
	e		0.309	0.844	0.786	0.481	0.540	**0.000**	**0.004**
	c			0.972	0.904	0.610	0.642	**0.000**	**0.008**
	g				0.865	0.656	0.501	0.360	0.166
	k					0.676	0.423	0.444	0.250
	i						0.890	0.468	0.277
	j							0.635	0.331
	h								0.340

	h	g	d	e	c
f	0.207	0.167	**0.000**	**0.000**	**0.000**
h		0.398	**0.000**	**0.000**	**0.000**
g			**0.000**	**0.000**	**0.000**
d				**0.012**	**0.005**
e					**0.047**

	h	g	e	d	c
f	**0.000**	**0.000**	**0.067**	**0.078**	**0.004**
h		0.555	0.305	0.321	**0.037**
g			0.374	0.389	**0.066**
e				0.536	**0.072**
d					**0.000**

F-test		e	c	g	k	i	j	h	f
	d	1.000	0.985	0.786	0.651	0.391	0.266	0.983	0.583
	e		0.984	0.786	0.651	0.391	0.266	0.983	0.583
	c			0.771	0.637	0.381	0.258	0.999	0.569
	g				0.856	0.558	0.400	0.770	0.781
	k					0.685	0.509	0.636	0.923
	i						0.798	0.380	0.757
	j							0.257	0.573
	h								0.569

	h	g	d	e	c
f	0.952	0.860	0.846	0.679	0.671
h		0.908	0.893	0.723	0.715
g			0.985	0.812	0.803
d				0.826	0.817
e					0.991

	h	g	e	d	c
f	0.966	0.986	0.224	0.285	0.276
h		0.569	**0.078**	0.106	0.102
g			**0.048**	**0.067**	**0.064**
e				0.321	0.310
d					0.140

Architecture: c) 6 H1 PE's d) 12 H1 PE's e) 18 H1 PE's f) 6 H1, 3 H2 PE's g) 12 H1, 4 H2 PE's h) 18 H1, 6 H2 PE's
i) 5 H1 PE's j) 10 H1 PE's k) 15 H1 PE's

Table E14c. Statistical comparison of narrow FOV NN3 network architectures.

FOV, mm	15																	
Features	ce									ef								
Network	m7n	m7j	m7c	m7d	m7e	m7i	m7l	m7m	m7k	m8c	m8i	m8k	m8m	m8j	m8n	m8l	m8d	m8e
Architecture	n	j	c	d	e	i	l	m	k	c	i	k	m	j	n	l	d	e
Mean error	-1.0	-0.8	-0.7	-0.4	-0.4	**0.0**	0.1	0.3	2.8	-0.1	-0.1	-0.1	**0.0**	0.1	0.3	0.5	0.7	0.7
Stdev error	19.8	15.1	19.8	19.9	20.2	**11.6**	17.5	16.2	12.0	12.6	12.6	**12.3**	12.7	12.5	12.5	13.5	12.9	13.1

t-test		j	c	d	e	i	l	m	k		i	k	m	j	n	l	d	e
	n	0.882	0.775	0.642	0.654	0.692	0.413	0.452	0.141	c	0.940	0.926	0.803	0.539	0.228	0.449	0.210	0.218
	j		0.958	0.857	0.847	0.709	0.619	0.390	**0.042**	i		1.000	0.858	0.370	0.286	0.439	0.267	0.286
	c			0.013	0.651	0.801	0.562	0.593	0.187	k			0.787	0.495	**0.070**	0.439	0.235	0.253
	d				0.973	0.873	0.691	0.691	0.220	m				0.823	**0.042**	0.529	0.327	0.343
	e					0.880	0.674	0.685	0.214	j					0.385	0.561	0.377	0.397
	i						0.952	0.895	**0.019**	n						0.805	0.598	0.622
	l							0.922	0.266	l							0.855	0.871
	m								0.189	d								0.936

F-test		j	c	d	e	i	l	m	k		i	k	m	j	n	l	d	e
	n	**0.061**	0.994	0.977	0.874	**0.000**	0.391	0.168	**0.001**	c	0.992	0.891	0.954	0.959	0.987	0.620	0.848	0.749
	j		**0.060**	**0.057**	**0.043**	**0.072**	0.306	0.617	0.110	i		0.883	0.961	0.952	0.980	0.627	0.856	0.756
	c			0.983	0.880	**0.000**	0.387	0.166	**0.001**	k			0.845	0.931	0.903	0.527	0.742	0.648
	d				0.897	**0.000**	0.375	0.159	**0.001**	m				0.913	0.941	0.662	0.894	0.793
	e					**0.000**	0.310	0.125	**0.000**	j					0.972	0.585	0.808	0.711
	i						**0.005**	**0.022**	0.840	n						0.609	0.836	0.737
	l							0.599	**0.009**	l							0.761	0.861
	m								**0.037**	d								0.898

Architecture: c) 6 H1 PE's d) 12 H1 PE's e) 18 H1 PE's
i) 5 H1 PE's j) 10 H1 PE's k) 15 H1 PE's l) 5 H1, 3 H2 PE's m) 10 H1, 5 H2 PE's n) 15 H1, 7 H2 PE's

Table E15a. Statistical comparison of narrow FOV NN3 network field of view.

Features					abc				
Architecture	7 hidden PE's			14 hidden PE's			21 hidden PE's		
Network	m1	m2	m3	m1a	m2a	m3a	m1b	m2b	m3b
FOV, mm	15	29	43	15	29	43	15	29	43
Mean error	**-2.2**	-2.7	-2.3	-2.0	-0.7	**-0.6**	-1.3	**-1.1**	-1.4
Stdev error	13.2	13.1	**12.5**	12.5	10.8	**10.7**	12.9	**10.3**	10.5
t-test		29	43		29	43		29	43
	15	0.242	0.899	15	0.221	0.201	15	0.887	0.888
	29		0.481	29		0.290	29		**0.056**
F-test		29	43		29	43		29	43
	15	0.943	0.687	15	0.303	0.270	15	0.109	0.150
	29		0.740	29		0.942	29		0.866

Features					bc				
Architecture	6 hidden PE's			12 hidden PE's			18 hidden PE's		
Network	m1c	m2c	m3c	m1d	m2d	m3d	m1e	m2e	m3e
FOV, mm	15	29	43	15	29	43	15	29	43
Mean error	**0.6**	-1.3	-3.4	0.4	**0.3**	-2.1	**0.3**	-0.6	-3.0
Stdev error	13.4	13.0	**12.7**	**12.6**	12.8	13.7	13.6	**12.8**	13.4
t-test		29	43		29	43		29	43
	15	**0.092**	**0.000**	15	0.820	**0.005**	15	0.421	**0.000**
	29		**0.000**	29		**0.003**	29		**0.002**
F-test		29	43		29	43		29	43
	15	0.812	0.718	15	0.917	0.549	15	0.710	0.937
	29		0.901	29		0.620	29		0.770

Features					bc				
Architecture	6 H1, 3 H2 PE's			12 H1, 4 H2 PE's			18 H1, 6 H2 PE's		
Network	m1f	m2f	m3f	m1g	m2g	m3g	m1h	m2h	m3h
FOV, mm	15	29	43	15	29	43	15	29	43
Mean error	1.7	**0.3**	-1.0	3.9	**-0.1**	-2.2	2.7	**0.0**	-2.7
Stdev error	13.4	**12.6**	12.7	**12.3**	12.5	14.2	13.6	12.7	**12.5**
t-test		29	43		29	43		29	43
	15	0.121	**0.013**	15	**0.000**	**0.000**	15	**0.003**	**0.000**
	29		**0.009**	29		**0.046**	29		**0.000**
F-test		29	43		29	43		29	43
	15	0.666	0.712	15	0.900	0.318	15	0.630	0.571
	29		0.950	29		0.382	29		0.933

Table E15b. Statistical comparison of narrow FOV NN3 network field of view.

Features					acd				
Connectivity	7 hidden PE's			14 hidden PE's			21 hidden PE's		
Network	m4	m5	m6	m4a	m5a	m6a	m4b	m5b	m6b
FOV, mm	15	29	43	15	29	43	15	29	43
Mean error	**-1.6**	-2.0	-2.0	-1.0	1.2	**-0.7**	-2.2	**-0.2**	-1.6
Stdev error	12.7	11.7	**10.7**	13.1	12.8	**10.7**	13.4	12.6	**10.6**
t-test		29	43		29	43		29	43
	15	0.691	0.803	15	**0.043**	0.782	15	0.250	0.587
	29		1.000	29		0.211	29		0.340
F-test		29	43		29	43		29	43
	15	0.571	0.246	15	0.888	0.166	15	0.674	0.106
	29		0.552	29		0.213	29		0.231

Features					cd				
Connectivity	6 hidden PE's			12 hidden PE's			18 hidden PE's		
Network	m4c	m5c	m6c	m4d	m5d	m6d	m4e	m5e	m6e
FOV, mm	15	29	43	15	29	43	15	29	43
Mean error	**0.5**	-3.1	-5.1	**-0.2**	-1.9	-3.6	**0.2**	-2.5	-3.6
Stdev error	13.4	12.2	**10.8**	13.4	12.6	**10.9**	13.4	12.2	**10.6**
t-test		29	43		29	43		29	43
	15	**0.000**	**0.001**	15	**0.096**	**0.056**	15	**0.004**	**0.028**
	29		0.139	29		0.241	29		0.448
F-test		29	43		29	43		29	43
	15	0.512	0.135	15	0.685	0.146	15	0.532	0.110
	29		0.400	29		0.293	29		0.327

Features					cd				
Connectivity	6 H1, 3 H2 PE's			12 H1, 4 H2 PE's			18 H1, 6 H2 PE's		
Network	m4f	m5f	m6f	m4g	m5g	m6g	m4h	m5h	m6h
FOV, mm	15	29	43	15	29	43	15	29	43
Mean error	2.9	**0.1**	-1.2	0.6	**-0.4**	-2.7	2.1	**-0.1**	-2.3
Stdev error	**12.4**	13.0	13.1	12.9	12.7	**12.6**	13.4	12.9	**12.8**
t-test		29	43		29	43		29	43
	15	**0.000**	**0.000**	15	0.552	**0.026**	15	**0.016**	**0.000**
	29		**0.009**	29		**0.000**	29		**0.000**
F-test		29	43		29	43		29	43
	15	0.735	0.702	15	0.908	0.894	15	0.770	0.714
	29		0.965	29		0.986	29		0.941

Table E16. Statistical comparison of wide and small FOV NN3 networks having similar architectures and input features.

Features	a0	cd			a0	cd			a0	cd		
FOV, mm	140	14	28	42	140	14	28	42	140	14	28	42
Network	w1	m4c	m5c	m6c	w1a	m4d	m5d	m6d	w1b	m4e	m5e	m6e
Mean error, mm	**-0.3**	0.5	-3.1	-5.1	4.0	**-0.2**	-1.9	-3.6	-1.2	**0.2**	-2.5	-3.6
Stdev error, mm	27.8	13.4	12.2	**10.8**	25.7	13.4	12.6	**10.9**	34.2	13.4	12.2	**10.6**
t-test		0.840	0.503	0.254		0.300	0.148	**0.063**		0.757	0.792	0.636
F-test		**0.000**	**0.000**	**0.000**		**0.000**	**0.000**	**0.000**		**0.000**	**0.000**	**0.000**

Features	a5	ef	a5	ef	a5	ef
FOV, mm	140	14	140	14	140	14
Network	w2	m8c	w2a	m8d	w2b	m8e
Mean error, mm	2.9	**-0.1**	1.9	**0.7**	2.8	**0.7**
Stdev error, mm	**11.9**	12.6	16.5	**12.9**	15.1	**13.1**
t-test		0.205		0.723		0.462
F-test		0.718		**0.088**		0.330

VITA

Michael Rigney

Candidate for the Degree of

Doctor of Philosophy

Dissertation: **NEURAL NETWORK FEATURE RECOGNITION: LOCATING THE CONIFER SEEDLING ROOT COLLAR IN A DIGITAL IMAGE**

Major Field: Biosystems Engineering

Biographical:

Personal Data: Born in Los Angeles, California, on May 16, 1960, the son of Sylvester and Denise Rigney. Married Michelle Lea Rollins in Stillwater, Oklahoma, on April 22, 1995. Son Alexander Mitchell born in Stillwater, Oklahoma, on February 24, 1996.

Education: Graduated from St. Bernard High School, Playa del Rey, California, in May 1978; received Associate of Technology degree from Seminole Junior College, Seminole Oklahoma in May 1983; received Bachelor of Science and Master of Science degrees in Agricultural Engineering from Oklahoma State University in May 1985 and December 1986, respectively. Completed the requirements for the Doctor of Philosophy with a major in Biosystems and Agricultural Engineering at Oklahoma State University in July, 1997.

Experience: Employed by Oklahoma State University, Agricultural Engineering Department as undergraduate and graduate research assistant, 1985-1986. Employed by International Robomation/Intelligence as machine vision applications engineer, 1987. Employed by Oklahoma State University, Biosystems and Agricultural Engineering Department as research engineer, 1988 to present.

Professional Associations: American Society of Agricultural Engineers, SPIE-Society of Photo-Optical Instrumentation Engineers, Alpha Epsilon (ASAE honorary), Sigma Xi (The Scientific Research Society).