Geoderma, 53 (1992) 237–253
Elsevier Science Publishers B.V., Amsterdam

Application of artificial neural system algorithms to image analysis of roots in soil, I. Initial results

Edward A. Nater [a], Keith D. Nater [b] and John M. Baker [c]

[a] *Soil Science Department, University of Minnesota, St. Paul, MN 55108, USA*
[b] *Science Applications International Corporation, San Diego, CA 92121, USA*
[c] *USDA-ARS, Soil Science Department, University of Minnesota, St. Paul, MN 55108, USA*

(Received January 15, 1991; accepted after revision March 26, 1991)

ABSTRACT

Nater, E.A., Nater, K.D. and Baker, J.M., 1992. Application of artificial neural system algorithms to image analysis of roots in soil, I. Initial results. In: A.R. Mermut and L.D. Norton (Editors), Digitization, Processing and Quantitative Interpretation of Image Analysis in Soil Sciences and Related Areas. Geoderma, 53: 237–253.

Several back propagation artificial neural system (ANS) architectures were tested to determine their abilities to identify roots in minirhizotron images of soil. The general model had an input level, which consisted of two linear arrays 1×11 pixels each, a hidden level, with 7 to 11 nodes, and an output level consisting of a single node set to produce a binary root/not-root output. The inputs to the model consisted of two linear arrays, one each from a horizontal and a vertical derivative image produced from the raw image by the Savitzky-Golay algorithm. A training image was constructed by hand-editing the raw image. The back propagation model was trained by repeatedly presenting it with a set of inputs and an associated target response. The error was calculated for the output from each input/target response pair and corrections to the weighting functions were made using a gradient descent back correction algorithm. The results of this study suggest that the ANS approach has potential to identify roots in images of soil. Suggestions for improving the performance of the model are presented.

INTRODUCTION

Minirhizotron research

A minirhizotron is a system of clear glass or plastic access tubes placed in the soil that can be used to view plant root systems. Most current minirhizotron installations are equipped with a small video camera or a 35-mm still camera to collect images. Root lengths, diameters, and rooting densities can be measured on minirhizotron images, thus allowing quantitative compari-

Correspondence to: E.A. Nater, Soil Science Department, University of Minnesota, St. Paul, MN 55108, USA.

sons of root response to varying soil management schemes, environmental parameters, or other experimental treatments. A minirhizotron system provides easy, rapid, non-destructive sampling along horizontal or nearly vertical transects through the soil, allowing a scientist to take a large number of images in a relatively short period of time. However, the amount of time required to manually measure root dimensions in these images is considerable and currently constitutes the main drawback to use of the system. Because root distributions are often spatially quite variable (Tardieu, 1988), particularly in well-structured soils, and the minirhizotron image represents a small (e.g., 12 by 16 mm for videotaped images) cross-section through the soil, a fairly large number of images may be required to assure statistical significance of results. Development of a method to automate the identification and measurement of roots in minirhizotron images would greatly enhance the utility of this type of research.

The difficult part of this task is to automate the identification of roots in minirhizotron images. This is a problem in feature recognition. Once identified, the automated measurement of root dimensions in binary (root, non-root) digitized images presents a less difficult, though not necessarily easy, task (Lebowitz, 1988).

Traditional methods of feature recognition

The recognition of root features in a minirhizotron image is a problem of separating foreground objects (roots) from the image background. Using traditional image processing techniques, a number of global and localized spatial transformations can be applied to an image to increase the spectral separation of foreground and background objects.

Global techniques

Thresholding is a global technique used to produce a binary image wherein all light intensity values within a specified range are assigned a new, single intensity value, and all values outside the range are assigned another. If the features of interest are either consistently lighter or darker than the other portions of the image, thresholding can be used for feature identification. Thresholding is successful when the distribution of light intensity values for objects in the desired feature space is distinct from that of all other objects. Intensity histograms of such images are inherently multi-modal.

At first glance, a minirhizotron image appears to be suitable for thresholding, as the roots are perceived to be lighter than the background in nearly all portions of the image (Fig. 1a). The resulting light intensity histogram (Fig. 1b), however, is not bimodal as expected. Many background features share the same light intensity distribution as a root. Human perception, therefore,

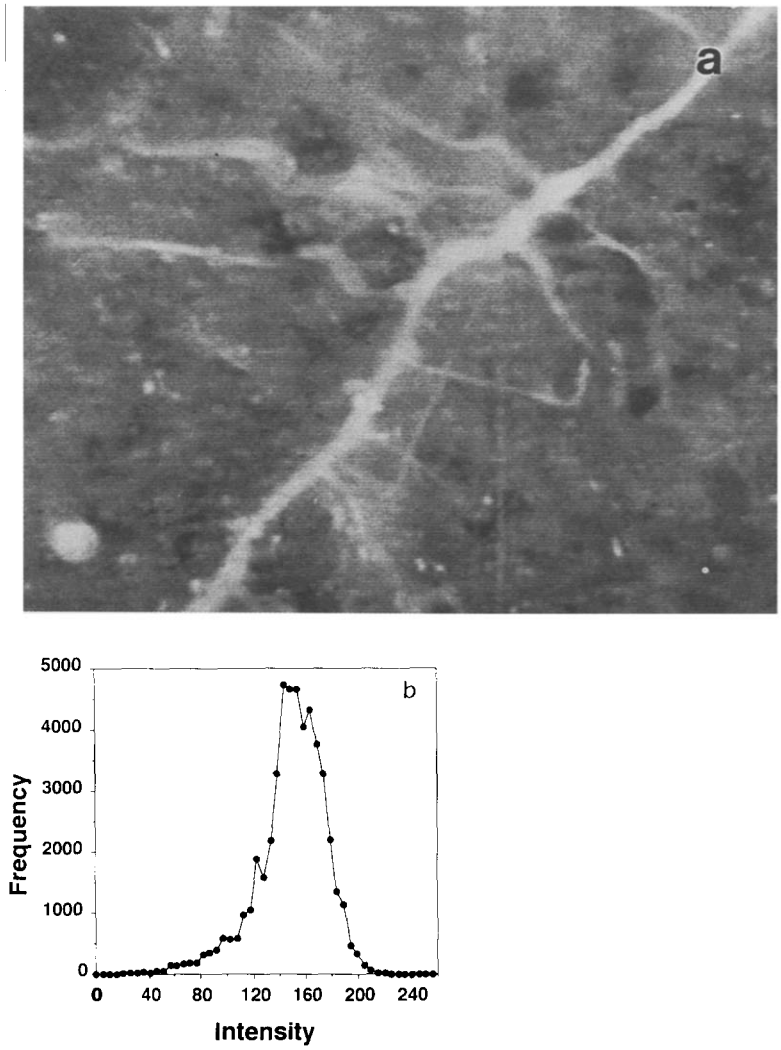

Fig. 1. A videotaped minirhizotron image (a) and its associated light-intensity histogram (b). Note the absence of a bimodal distribution in the intensity histogram.

must use other, localized clues to distinguish roots from the remaining background.

Local techniques

Edge tracing, or outlining, is an example of a spatially localized transformation used in feature recognition. This technique employs a convolution kernel to delineate areas of steep localized change in light intensity, thereby outlining features that are either much brighter or darker than the surrounding portions of an image. Once an initial set of edges has been detected, other

Fig. 2. The result of outlining the raw image presented in Fig. 1a, showing outlining of numerous features other than roots.

transformations are applied to grow continuous edges from edge segments and to skeletonize thick edges to a uniform size.

If variations in image light intensity are viewed as one dimension of a three-dimensional surface, then edge tracing is a global transformation, thresholding, applied to the first derivative of the image. Points with tangents exceeding a specified slope are assigned one intensity value and all other points are assigned another. Edge tracing will succeed only if the distribution of first derivative values for the edges of the desired feature is distinct from the distribution for all other edges in the image.

Roots exhibit large, localized changes in light intensity, but analysis reveals that other features, such as pores and cracks in the soil, exhibit the same rate of change of intensity as do roots (Fig. 2). Thus, outlining algorithms alone are also unable to accurately and exclusively delineate roots.

Rule-based techniques

Rule-based, or expert system, methods recently have been applied to problems of feature identification, and more particularly to the identification of roots in minirhizotron images (Ferguson et al., 1990; Stockman et al., 1990). Expert system methods use codified rules based on intensity, contrast, and root morphological characteristics (such as length-to-width ratios or orientation of axes) to identify features (Stockman et al., 1990). They require the programmer to codify rules that can distinguish between the features in an image. It is difficult to formulate such rules a priori, and empirical develop-

ment of them often leads to complex structures in an attempt to allow for all situations observed. Rule-based methods have met with success on well-resolved minirhizotron images (Stockman et al., 1990), but may be prone to gross misjudgements when applied to noisier or more complex images.

Introduction to artificial neural systems

Artificial neural systems (ANS) are a class of mathematical models proposed to explain how neurons in higher organisms can interact to store and retrieve information (McClelland et al., 1986). Although a complete description of their theory and use is beyond the scope of this article, a brief description of the architecture and operation of an ANS is given to acquaint the reader with the methods used in these experiments. For a more complete description the reader is referred to Rumelhart et al. (1986a) and Wasserman (1989).

Artificial neural systems are used to solve pattern recognition problems because they can be trained to classify input patterns into user-defined categories, even in the presence of significant noise. The network learns to classify patterns by example, not on the basis of predetermined rules. The ANS is trained by presenting a pattern and expected response to the network. The network will make an initial classification for the pattern and then adjust its internal representation to reduce the error between this initial classification and the expected answer. This ability to *self-organize*, even in the presence of noise, is a significant advantage over rule-based systems. Computer implementations of artificial neural systems are generally simple in concept and easy to code. They are, however, computationally intensive and generally require the use of high-speed computers with several megabytes of dynamic memory.

Artificial neural systems are organized as a hierarchy of two or more layers or slabs. The model used in this study, called a back propagation model, has three levels referred to as the input layer, the hidden layer, and the output layer (Fig. 3). Nodes in the input layer are connected to each node in the hidden layer. The hidden layer is similarly connected to the output layer. Each connection has a weight or connection strength associated with it which governs the lower-level node's contribution to the higher-level node's input or activation.

In a computer implementation of a back propagation network, nodes in the input level are loaded with floating point values representing the pattern to be classified. For each connection to a node in the hidden layer, each input value, O_i, is multiplied by an independent weighting function, w_{ji}. The weighted values are then summed for each node in the hidden layer:

$$net_j = \sum_i w_{ji} O_i \qquad (1)$$

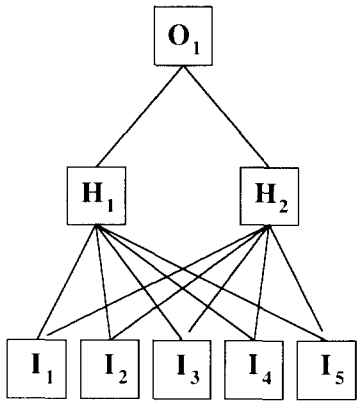

Fig. 3. Generalized architecture of an artificial neural system. Boxes are nodes in the input (I), hidden (H), and output (O) layers.

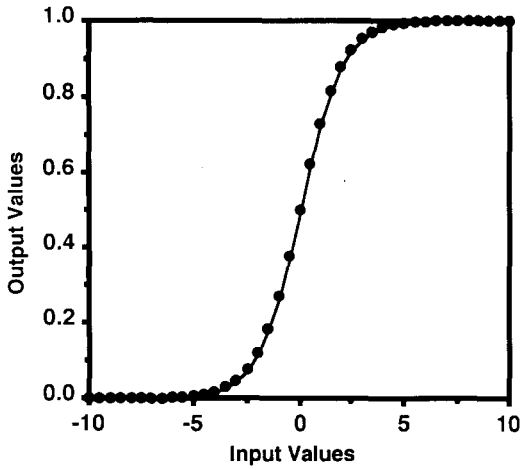

Fig. 4. Output of the sigmoid function with Θ_j equal to zero.

where net_j is the input to any individual hidden node, and the subscripts i and j refer to nodes in the input and hidden layers, respectively.

The sum of all weighted inputs to each node in the hidden level is passed through a nonlinear logistic activation (sigmoid) function:

$$O_j = \frac{1}{1+\exp[-(net_j+\Theta_j)]} \tag{2}$$

where Θ_j is a bias similar in function to a threshold. The sigmoid function returns values close to either zero or one for most values of net_j (Fig. 4).

The same weighting and summation processes occur between each node of the hidden level and the node(s) of the output level (subscript k). The sum

of all inputs to each node of the output level, net_k, is also passed through the semi-linear activation function (eq. (2)), thus producing the final output of the back propagation model, O_k, which resembles a binary switch, in that the values output tend to be close to either zero or one.

During initialization, small random numbers are chosen for each of the weighting factors. Obviously, the first result from the back propagation model will most likely be incorrect, and the weights will have to be adjusted to obtain a more correct result. This process is referred to as the training or learning phase. During the training phase, the output of the back propagation model is compared to a target result supplied by the programmer. The error, E, is then calculated for any input/output pair p by:

$$E=\frac{1}{2}\sum_{k}(t_{pk}-O_{pk})^2 \tag{3}$$

where $(t_{pk}-O_{pk})$ is δ_{pk}, the difference between the expected result, or training value, and the value output by the back propagation model. A gradient descent back propagation process then adjusts each set of weights by $\eta\delta_{pk}j_{pk}$, where η is a constant, the learning rate, and j_{pk} is the value of the jth element of the hidden level. This type of ANS is commonly referred to as a delta rule back propagation model. The error function is then propagated back to the hidden level and a similar process is used to correct the weights operative between the input level and the hidden level. The basic algorithm described here has been enhanced by the use of learning rate and momentum terms to avoid local minima (Rumelhart et al., 1986b, pp. 329–330).

Overall, the training process is an attempt to perform least squares minimization of the total error of the network. Over the course of hundreds or thousands of iterations, the ANS often, though not always, converges gradually on a set of weights that minimizes the sum of squares error and provides a solution to the problem at hand.

Artificial neural systems have been used in a variety of pattern recognition and conversion problems. Networks have been trained to convert text to speech (Sejnowski and Rosenberg, 1987), to quantify and calibrate spectroscopic data (Long et al., 1990), and for optical recognition of characters (Rajavelu et al., 1989).

In view of the limitations of other techniques and the potential of ANS, the objective of this study was to develop ANS algorithms to automatically identify roots in minirhizotron images of soil.

MATERIALS AND METHODS

Minirhizotron photography

Polybutyrate access tubes were installed horizontally at three depths in a field corn (*Zea mays* L.) plot located on a Waukegan silt loam (Typic Haplu-

doll) at the Rosemount Agricultural Experiment Station of the University of Minnesota, located near Rosemount, Minn., U.S.A (Pettygrove et al., 1988). Minirhizotron images (12 by 16 mm) were taken using unfiltered visible light by a Circon Microvideo camera (Circon Corp., Santa Barbara, CA 93111, U.S.A.). Root diameters were commonly 0.1 to 0.5 mm. During videotape recording, minirhizotron images were viewed on a monitor to allow proper orientation of the camera in the access tubes.

Image digitization

Images were captured from videotape in 256 levels of gray by a Scion FG-2 video capture board (Scion Corp., Walkersville, MD 21793, U.S.A.) using its associated software, Video Image 1000 v. 3.32. The video capture board was installed in a Macintosh II computer. Image brightness and contrast were adjusted visually to operator satisfaction before image capture. Original images were 500 pixels wide by 400 pixels high. In order to reduce the computational time required to process each image, reduced-size images were produced by discarding alternate rows and columns of pixels, thus yielding images 250 pixels wide by 200 pixels high.

Production of derivative images

As illustrated earlier (Fig. 2), gray-scale thresholding did not satisfactorily identify roots in these images, even though the roots appeared visually to be brighter than soil materials in nearly all of these images. This implies that the relative brightness of a root is probably more important for visual recognition than its absolute brightness. Consequently, first derivatives of the raw image (Fig. 5a) were produced along both horizontal and vertical transects (Figs. 5b, 5c) using the Savitzky-Golay algorithm (Savitzky and Golay, 1964; Steiner et al., 1972). Several different fits were tried; a seven-point quadratic fit appeared to be the best compromise between root edge sharpness and random noise level. A nine-point cubic fit gave visually similar results, but it also eliminated one extra row of pixels along each border of the image. Before being input to the back propagation model, the derivative values were scaled to the interval 0.0 to 1.0 by dividing by a constant close to the maximum observed value and adding 0.5 to the result. The values 0.0 and 1.0 were substituted for scaled values < 0.0 and > 1.0, respectively. The horizontal and vertical derivative images were then used as inputs to the ANS, both during the training sessions and during analysis.

As expected, clear differences were observed between the horizontal and vertical derivative images (Figs. 5b, c). Because horizontal roots were barely visible in the horizontal derivative image, but were easily recognized in the

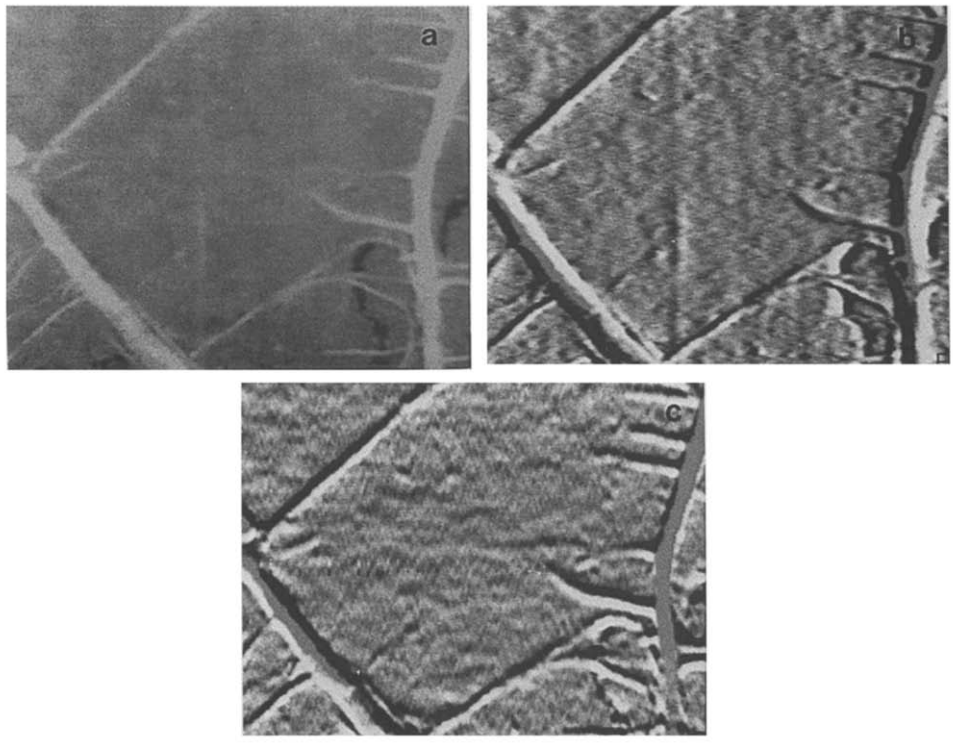

Fig. 5. Raw image (a), and the horizontal (b) and vertical (c) derivative images produced from it by the seven-point quadratic Savitzky–Golay smoothing algorithm.

vertical derivative image, and vice versa, both derivative images were used as inputs to the ANS.

Production of the training image

A binary training image (Fig. 6) was produced by a combination of hand-editing and thresholding of a gray-scale image using Image v. 1.31p (Rasband, 1990) on a Macintosh II computer. The gray-scale image was enlarged from 3 to 8 times and editing was conducted on a pixel-by-pixel basis. Once the background material had been colored black, the image was thresholded to produce a binary image.

Even though the image being edited was visually compared with its corresponding gray-scale image during the editing process, it was not always clear to the editor exactly which pixels were part of the root and which were not. Although the editor tried to be as consistent as possible during the production of the training image, some arbitrary decisions were made. For linear features like roots, which have large perimeter to area ratios, edge pixels constitute a

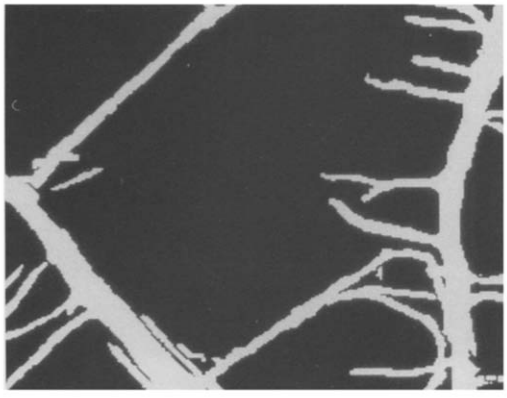

Fig. 6. Binary training image produced by hand-editing the image in Fig. 5a.

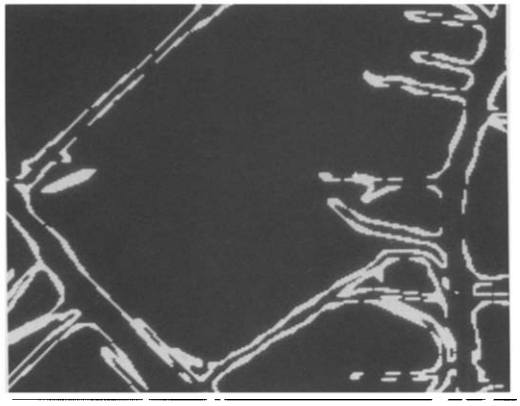

Fig. 7. The disallowed pixel map.

sizable portion of all pixels associated with roots, in this case perhaps as much as 20%. Although back propagation networks often have the ability to handle noisy data, the authors felt that the presence of such a high proportion of potentially arbitrary training values might create problems in the use of the binary image as the training image because they would send mixed signals to the ANS and enforce improper corrections. This can produce poor or slowed performance of the ANS, or even a lack of convergence.

In order to avoid problems with the use of arbitrary training data, the ANS was not trained on suspect pixels near the root–soil border. A map of disallowed pixels was produced by the following process. The binary training image was first subjected to a digital *erosion* process which removed two layers of pixels from the border of the root. The original binary training image was also subjected to a digital *dilation* process, which added two layers of pixels to the original border of the root. A third image, the disallowed pixel map,

was produced by subtracting the eroded image from the dilated image (Fig. 7). All pixels associated with non-zero values in the disallowed pixel map were excluded from the training set. Because the ANS was not trained on pixels along the root–soil border, it was uncertain whether the ANS would identify them as root or soil; however, it was assumed that whatever choice was made, it would be consistent, which was our main concern.

Architecture of the ANS

A program to produce a delta rule, back propagation ANS was written in the C language for execution on a Sun Sparcstation 1+ workstation with 16 megabytes of dynamic RAM. Inputs to the ANS consisted of 11 pixels each in the vertical and horizontal derivative images (Fig. 8), with the midpoint of each 11-pixel array centered on the pixel of interest. The number of hidden nodes was varied between 7 and 11. Only one output node was used which corresponded to a binary decision on whether the pixel of interest was part of a root. All 22 input nodes were connected to each node in the hidden layer, and all nodes in the hidden level were connected to the output node.

During the training phase, a result was calculated for a single pixel, compared to the training value, and corrections were applied. The ANS was moved systematically across and down the image on a pixel-by-pixel basis, excluding those pixels that were identified in the disallowed pixel map. The results for each pixel were compared to their corresponding values in the training image, and corrections were applied.

At every tenth iteration through the training image, a mean error for the whole image was calculated and the current weights were saved in a separate file. Once the mean error reached an apparently stable minimum, the training

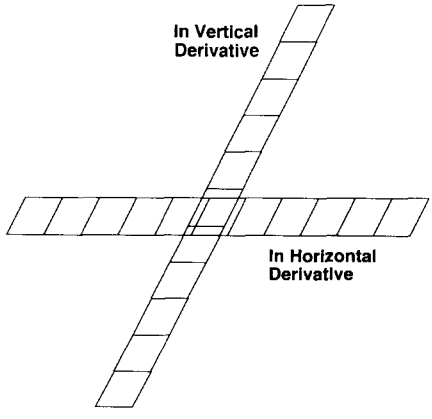

Fig. 8. Inputs to the model consisted of 11 pixels each in the vertical and horizontal derivative image, each centered on the pixel of interest.

phase was stopped. Identification of roots in the image was then attempted on the training image and other images using the saved weights.

RESULTS AND DISCUSSION

Decrease in mean error

For a typical training session, the mean error showed an overall decrease with increasing number of iterations (Fig. 9). Fluctuations of the mean error in the early part of the training phase were usually associated with movement in and out of local minima on the response surface. Although the gradient descent algorithm included learning and momentum terms, the network would occasionally become trapped in a local minimum and have to be restarted with a different set of random weights in order to avoid that particular minimum. Typically, the first local minimum encountered is a weight configuration that gives non-root values for all pixels. This configuration represents a local minimum because the majority of pixels in the image are associated with non-root features. Other local minima are associated with more complex properties of the image or with recognition of some roots by specific properties that may not be shared by all roots.

Results on the same image

Results of a single training session using a network architecture with 7 hidden layer nodes are displayed in Figs. 10a–10d for 10, 20, 30, and 40 iterations through the training image, respectively. This sequence displays an overall, though not necessarily a monotonic, increase in the ability of the net-

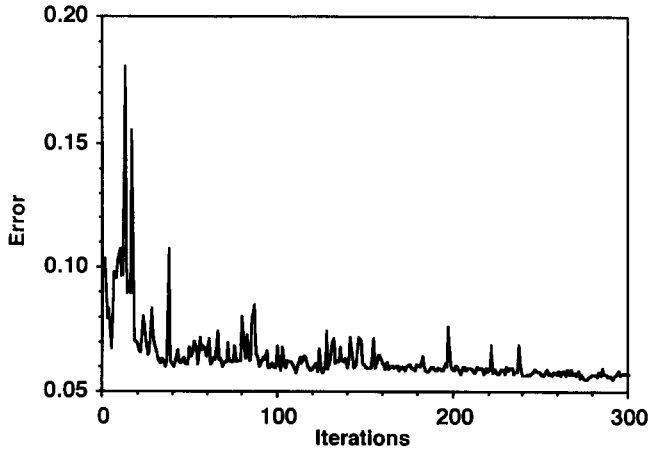

Fig. 9. Relationship between mean error and iteration during the learning phase.

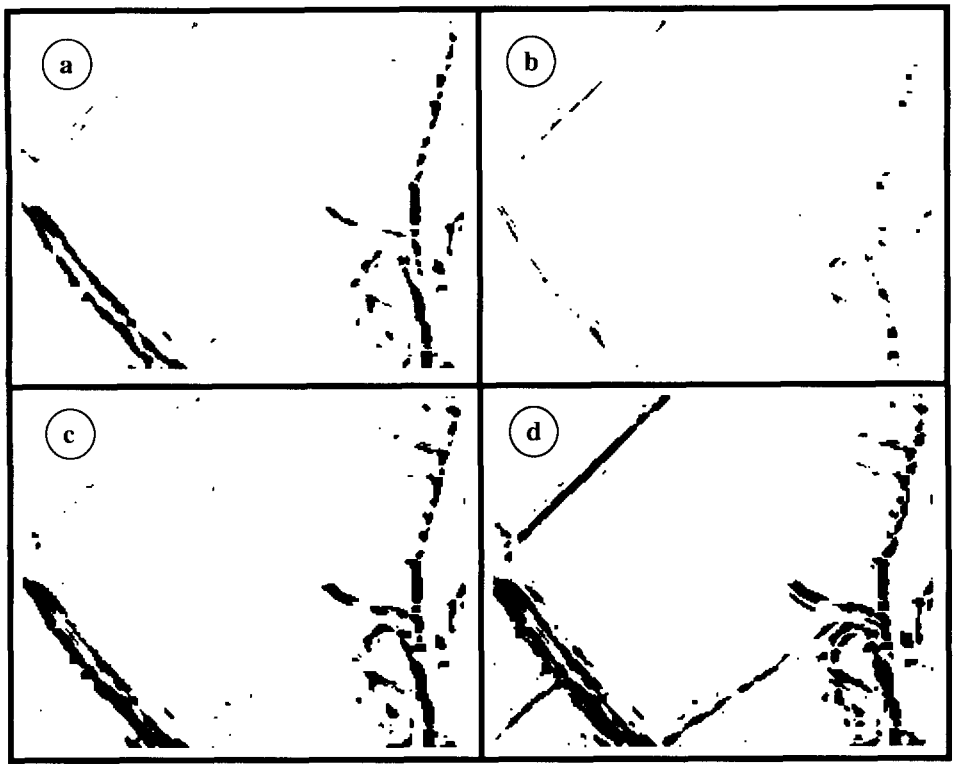

Fig. 10. Results on the training image at 10 (a), 20 (b), 30 (c), and 40 (d) iterations.

work to recognize roots and a progressive refinement of the ability of the ANS to discriminate against other features in this image. Though not entirely accurate, the network correctly identified most of the pixels associated with roots.

The thickness of the roots identified by the ANS varied from image to image. As discussed earlier, the ANS was not trained on pixels near the root–soil border, and thus was allowed to consider them as either root or background without penalty or correction. Apparently these pixels were identified as roots in some instances (thick root image, Fig. 10d) and as soil at other times (thin root image, Fig. 10c).

Other problems of identification are also evident in this sequence. For example, in the lower left-hand quadrant of Fig. 10d, pixels in the middle of the thick root are identified as background, even though they are clearly within the root. The width and height of the root (12–16 pixels) exceeded the length of our input pattern (11 pixels) at this particular location on the image. Using the inputs given, it would be difficult for the ANS to distinguish the interior of a root (all white) from open areas of background (all black), because the first derivatives of both conditions are similar. Consequently, either a

longer input slice is required for identification of root pixels at sites like this or additional processing is required to eliminate "holes" in roots before root measurements are taken.

Results on other images

In order to be useful, the ANS must be able to identify roots in images other than the training image. In Fig. 11, several raw images are compared and the results predicted by the ANS for these images. The results are not as good as those obtained for the original training image. This is not surprising considering that these images contain features not present in the original training image. Although the majority of the roots in these images were correctly identified as roots, many non-root features were also identified as roots, indicating that the algorithm, as configured and trained, had converged on a solution to the identification of roots, but had not developed the resolution to discriminate between some of these other features and roots. Because the ANS converges on solutions during the training session, it is reasonable to assume that it may misinterpret features presented later in images that were not presented during the training phase.

Obviously, a first step at finding a solution to this problem was to include a broader range of characteristics and features in the training set. To this end, the network was presented three different images and associated training sets during the training phase. However, it did not attain a stable minimum during several hundred iterations through the images. Part of the problem encountered here may be that, by combining horizontal and vertical slices in the same input pattern, we have inadvertently produced an XOR (*exclusive–or*) logistic problem: "if the vertical slice indicates an edge or the horizontal slice indicates an edge, signal an edge." Although back propagation networks can solve XOR problems (Rumelhart et al., 1986b, pp. 319–322), it generally takes longer for them to converge and they often get stuck in local minima.

Variations in contrast and dynamic range between images may also have contributed to inconsistent feature recognition between images (Fig. 12). Although back propagation networks are noise-resistant, they are based on the premise that all patterns share a single representation scale, i.e., any particular value (e.g., 0.7) is assumed to represent the same change in light level across all images. If the patterns do not share the same representation scale, the networks may give inconsistent results and misidentify features. Transforming pixel intensity values to standard contrast and dynamic range before input to the ANS should reduce this problem.

Another possible problem is that the input pattern does not contain enough information to make recognition possible. The inclusion of a wider swath (e.g., a 3 by 11 slice from each derivative image) or longer pattern might make it easier for the network to recognize continuous edges. Using the input pattern

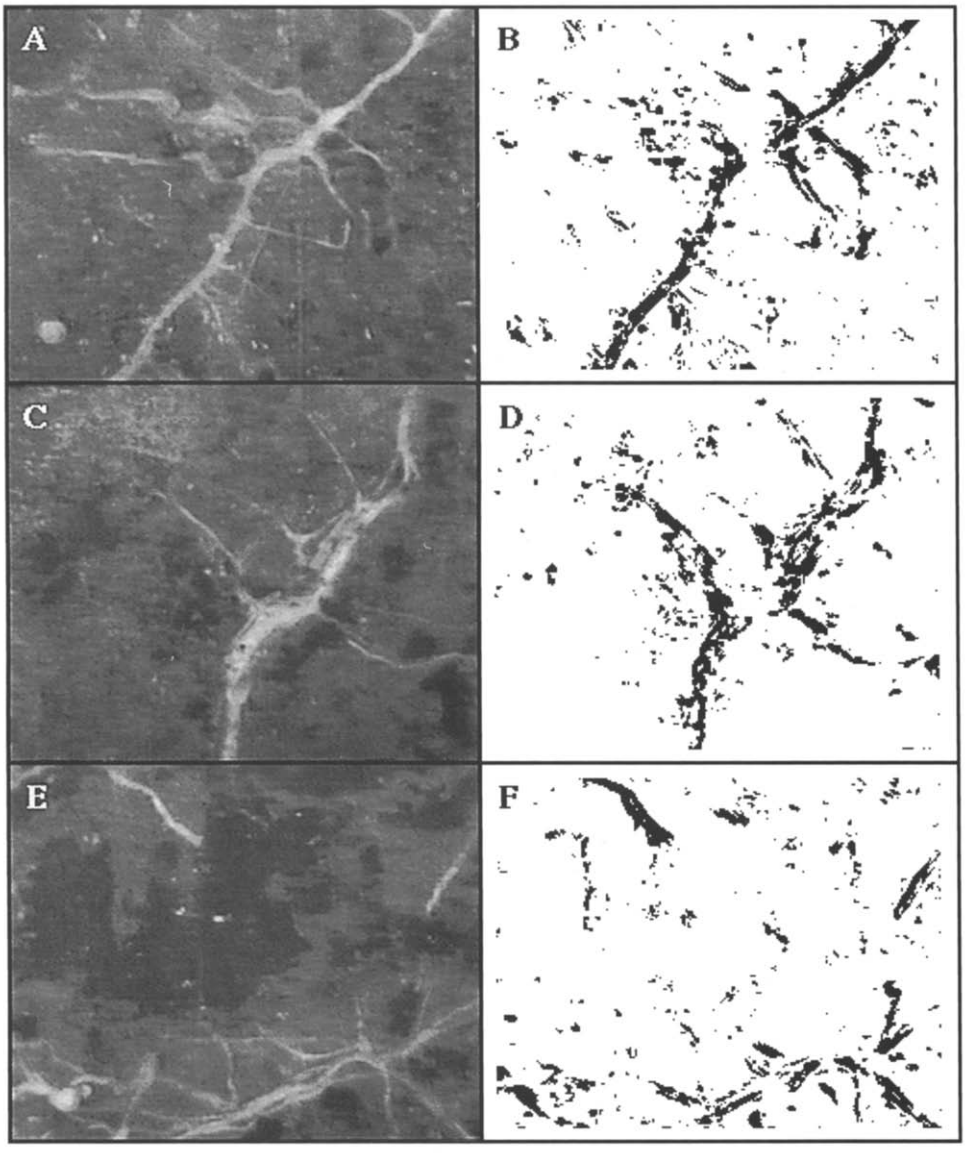

Fig. 11. Pairs of raw images (A, C, E) and the results calculated for them (B, D, F, respectively).

given, the network would have trouble developing a rule like "roots include a long adjacent series of high slopes" or even "roots are linear features". However, larger input patterns require more time for calculation of each iteration and may require more iterations to reach convergence. With the current architecture of 22 input layer nodes and 7 hidden layer nodes, one iteration through a 250×200 pixel image requires about 2 min on a Sun Sparcstation

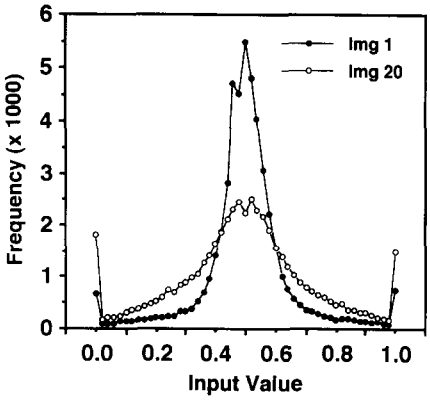

Fig. 12. Comparison of histograms for image 1 (Fig. 5a) and image 20 (Fig. 11e).

1+ workstation. Significantly larger network architectures will require more processing time, and may require more dynamic RAM storage as well. The increase in processing time is nonlinear because the number of connections increases faster than the number of nodes added.

CONCLUSIONS

Although the results reported in this manuscript are preliminary in nature, they suggest that the ANS approach has potential for the identification of roots in images of soil. The back propagation model presented in this paper was able to accurately identify most pixels in the training image correctly after approximately 40 iterations through the training set. However, there was a substantial decrease in accuracy when applied to other images on which it had not been trained, in part due to the presentation of features in these images that had not been presented to the ANS during the training phase.

This research is far from complete at this time. Future directions include testing different and more complex ANS architectures, particularly architectures containing larger input slices. Pre-processing techniques, such as contrast and histogram equalization, will be employed in order to make each set of inputs as similar as possible.

Research is also being conducted to try to produce higher-quality images in the field. Different wavelength filters for the minirhizotron camera light source are being tested to determine which wavelengths will increase contrast levels between the roots and the background.

ACKNOWLEDGEMENTS

The authors thank Drs. G. Stuart Pettygrove, Michael Russelle, and Ray Allmaras for supplying the minirhizotron videotape images, and Pete Becken for assistance in the early stages of model development.

Scientific Journal Series Paper No. 18752, Minnesota Agricultural Experiment Station, University of Minnesota, St. Paul, MN 55108, USA.

REFERENCES

Ferguson, J.C., Chang, S.L. and Smucker, A.J.M., 1990. Automated segmentation of minirhizotron images based on the coherence of intensity gradients. In: Abstr., Conf. Analytical Methods for Quantifying Root and Soil Dynamics, St. Louis, MO, August 13–14, p. 7.

Lebowitz, R.J., 1988. Digital image analysis measurement of root length and diameter. Environ. Exp. Bot., 28 (3): 267–273.

Long, J.R., Gregoriou, V.G. and Gemperline, P.J., 1990. Spectroscopic calibration and quantitation using artificial neural networks. Anal. Chem., 62: 1791–1797.

McClelland, J.L., Rumelhart, D.E. and Hinton, G.E., 1986. The appeal of parallel distributed processing. In: D.E. Rumelhart, J.L. McClelland and the PDP Research Group (Editors), Parallel Distributed Processing—Explorations in the Microstructure of Cognition, Vol. 1. Foundations. MIT Press, Cambridge, Mass., pp. 3–44.

Pettygrove, G.S., Russelle, M.P., Becken, P. and Burford, P., 1988. Number, surface area, and spatial distribution of corn roots in minirhizotron images. In: W.E. Larson (Editor), Mechanics and Related Processes in Structured Agricultural Soils. NATO Adv. Sci. Ser. E172, Kluwer, Dordrecht, p. 250.

Rajavelu, A., Musavi, M.T. and Shirvaikar, M.V., 1989. A neural network approach to character recognition. Neural Networks, 2 (5): 387–393.

Rasband, W., 1990. Image v. 1.31p. Public domain image processing software for the Macintosh. National Institute of Health, Research Services Branch, Washington, D.C.

Rumelhart, D.E., McClelland, J.L. and the PDP Research Group (Editors), 1986a. Parallel Distributed Processing—Explorations in the Microstructure of Cognition, Vol. 1. Foundations. MIT Press, Cambridge, Mass, 547 pp.

Rumelhart, D.E., Hinton, G.E. and Williams, R.J., 1986b. Learning internal representations by error propagation. In: D.E. Rumelhart, J.L. McClelland and the PDP Research Group (Editors), Parallel Distributed Processing—Explorations in the Microstructure of Cognition, Vol. 1. Foundations. MIT Press, Cambridge, Mass., pp. 318–330.

Savitzky, A. and Golay, M.J.E., 1964. Smoothing and differentiation of data by simplified least squares procedures. Anal. Chem., 36 (8): 1627–1639.

Sejnowski, T.J. and Rosenberg, C.R., 1987. Parallel networks that learn to pronounce English text. Complex Systems, 1:145–168.

Steiner, J., Termonia, Y. and Deltour, J., 1972. Comments on smoothing and differentiation of data by simplified least square procedure.8 Anal. Chem., 44: 1906–1909.

Stockman, G., Smucker, A.J.M. and Halgren, R., 1990. Processing images of complex root systems. In: Abstr., Conf. Analytical Methods for Quantifying Root and Soil Dynamics, St. Louis, MO, August 13–14, p. 18.

Tardieu, F., 1988. Analysis of the spatial variability of maize root density. Plant Soil, 107: 259–266.

Wasserman, P.D., 1989. Neural Computing Theory and Practice. Van Nostrand Reinhold, New York, N.Y., 230 pp.