

Progress in Analysis of Computed Tomography (CT) Images of Hardwood Logs for Defect Detection

Erol Sarigul^a, A. Lynn Abbott^a, and Daniel L. Schmoltd^b

^aBradley Department of Electrical and Computer Engineering,
Virginia Tech, Blacksburg, VA

^bUSDA Cooperative State Research, Education, and Extension Service, Washington, DC

Abstract

This paper addresses the problem of automatically detecting internal defects in logs using computed tomography (CT) images. The overall purpose is to assist in breakdown optimization. Several studies have shown that the commercial value of resulting boards can be increased substantially if defect locations are known in advance, and if this information is used to make sawing decisions. The problem is difficult, particularly for hardwood species, because of the natural variations of wood density and because of the irregular placement of defects such as knots. In our previous work, we developed a processing approach that utilizes artificial neural networks (ANN) to classify CT log images on a pixel-by-pixel basis. The system uses small (e.g., 5-by-5) neighborhoods in an image make a preliminary classification decision, using labels such as “knot,” “split,” and “bark.” This approach has yielded high accuracy statistically, with classification rates often exceeding 95%. Subjectively, however, the results can often be improved somewhat through further processing steps. For example, relatively simple operations can be employed to remove small, spurious regions that are not statistically significant, although their removal significantly improves the appearance of the results. This paper presents recent results for two hardwood species (red oak and sugar maple), as well as preliminary results for a softwood species (black spruce).

Introduction

Conventional practices for log processing, particularly during sawing operations, result in a considerable amount of waste. Improved methods are needed to assist the hardwood industry as it responds to increased demand for lumber in the face of decreasing forest resources. Several studies (e.g., [2, 3, 10, 14]) have shown that the commercial value of resulting boards can be increased substantially if sawing strategies are chosen using knowledge of internal defects

One method of obtaining information concerning the internal structure of logs is through computed tomography (CT) imaging. This nondestructive technique provides image “slices” that represent cross-sectional density distributions of a scanned object. Because of the large amount of data that is typically collected during CT scanning, it is desirable to have automated techniques that can quickly analyze the images to locate and identify defects, and to propose breakdown strategies.

Researchers at Virginia Tech and at the Southern Research Station of the USDA Forest Service have developed an approach that uses artificial neural networks (ANNs) to classify CT voxels individually, using small neighborhoods of CT density values as input feature vectors [7, 11, 12]. The ANN assigns a label such as “knot,” “decay,” or “clear wood” to each foreground pixel in the image. In several studies, the results have been quite good statistically, with classification accuracies often exceeding 95%.

In spite of the quantitative success of the approach, the resulting image classification from the ANN can often be refined through postprocessing. In some cases, for example, the ANN produces small spurious regions that do not seriously affect the statistical accuracy, but are qualitatively undesirable and may adversely affect subsequent analysis of the log. As another example, the ANN occasionally assigns incorrect labels to the outside border of the wood, because these portions of the log appear as low-density points due to spatial quantization effects. A postprocessing step can make use of global, domain-specific information to refine regions, thereby offsetting the ANN’s heavy reliance on local pixel neighborhoods.

There are several difficulties in designing a postprocessing module, however. Although many postprocessing steps can be easily implemented, different situations (possibly depending on the species of wood, on particular defect types, on the intended use of a log, and on personal preferences of the user) may require different types and degrees of region refinement. For these reasons, we are developing a system that can learn postprocessing rules automatically. The system observes the steps taken as a human user interactively edits a processed image, and then infers rules from those actions. During the system’s “learn mode,” the user views labeled images and makes refinements through the use of a keyboard and mouse. As the user manipulates the images, the system stores information related to those manual operations, and develops internal rules that can be used later for automatic postprocessing of other images. After one or more training sessions, the user places the system into its “run mode.” The system then accepts new images, and uses its rule set to apply postprocessing operations automatically in a manner that is modeled after those learned from the human user. At any time, the user can return to learn mode to introduce new training information, and this will be used by the system to update its internal rule set.

The system does not simply memorize a particular sequence of postprocessing steps during a training session, but instead generalizes from the image data and from the actions of the human user so that new CT images can be refined appropriately. Because it learns from a human “teacher,” this approach represents a form of supervised machine learning. However, the level of supervision is relatively mild by traditional machine-learning standards, because the teacher does not need to be knowledgeable concerning internal feature spaces or representations for rule selection. Because of its ability to accept new training inputs over time, the system is said to perform “incremental” (or “dynamic” or “on-line”) learning. This contrasts with many machine-learning systems, which require all training data to be made available at the beginning. Such systems perform “batch” (or “static”) learning.

The remaining sections of this paper provide an overview of the system, more discussion about the machine-learning approach, and results that we have obtained with the system.

System Overview

The overall classification system consists of three modules: (1) a preprocessing module, (2) an artificial neural network (ANN) module that performs tentative image segmentation, and (3) a postprocessing module. The preprocessing module distinguishes wood from background (air), and normalizes CT density values. The ANN module labels each non-background pixel of a CT slice using histogram-normalized values from small windows of size $3 \times 3 \times 3$ or 5×5 , centered on each pixel location to be classified. In the postprocessing module, morphological (shape-based) operations are applied to remove spurious regions and to refine region shapes.

Figure 1 illustrates the nature of the problem with an example image slice of a red oak log. In the figure, output from an ANN appears on the right without postprocessing. The largest region on the log section corresponds to clear (unblemished) wood. The ANN has correctly located a split near the center of the image, but has also incorrectly applied the split label along some of the nearby annual rings. A large knot region is present above and to the left of the split; this has an irregular shape that is difficult even for human observers to delineate precisely. For the knot, the ANN has produced several small regions that should be merged or removed for aesthetic reasons. Most bark pixels have been labeled correctly, but several other portions of the image have been incorrectly labeled as bark.

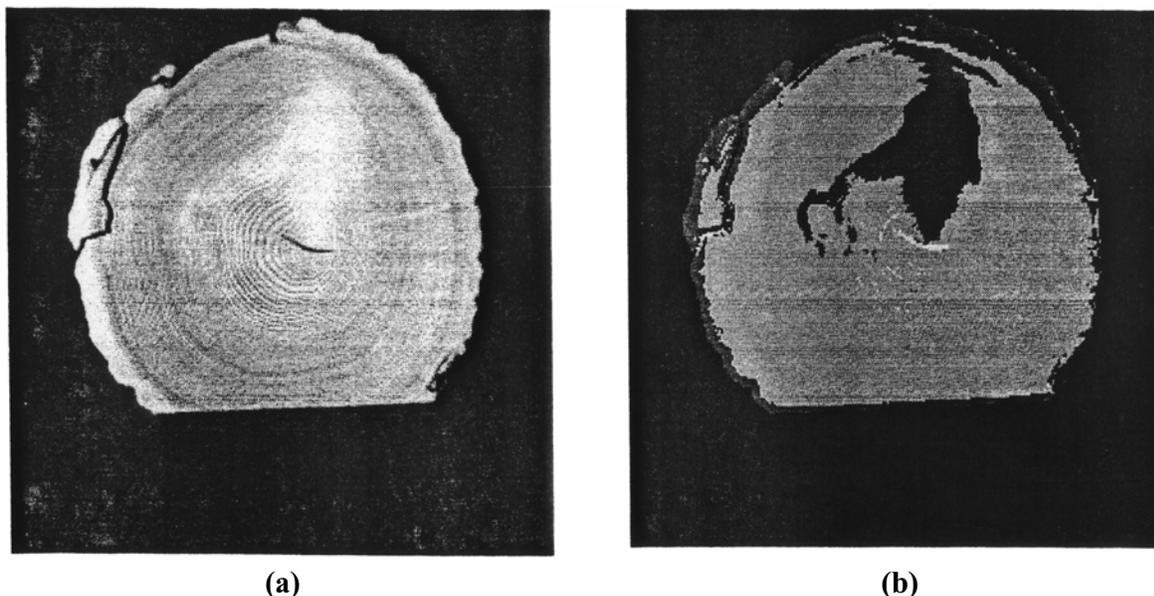


Figure 1. Output for a red oak log, without postprocessing. (a) CT slice to be analyzed. (b) Output of ANN classifier. For this example, we purposely used an over-trained neural network, which tends to generate output that is relatively low in accuracy.

Many of these incorrect labels have negligible effect on statistical classification accuracy, which depends on pixel counts alone. Qualitatively, however, the removal of several small regions and the smoothing of region contours can be desirable. Most of the needed changes

can be accomplished with relatively simple post processing steps. The difficulty lies in the development of rules that determine when to apply these simple steps. For example, the true split region near the center of Figure 1(b) is relatively small in size. A filter that indiscriminately removes all regions smaller than some threshold may also remove valid defect region such as this. Because it is difficult to manually specify an exhaustive set of rules that will work well for all possible situations, the emphasis of this research has been to allow the machine to develop its own rules, based on observations of a human user.

Interactive Learning Post processing System

Many supervised machine-learning systems, including the one described here, operate in two different modes: a “learn mode” and a “run mode”. These are illustrated in Figures 2 and 3, respectively. During the learn mode, our system provides a graphical user interface that allows a human user to edit segmented images. The user selects image operators from a menu, designates portions of the image to be processed, and observes the results. This interaction can continue until the user is satisfied with the resulting segmentation for any number of training images.

The menu of image-editing operators includes such region tasks as remove, fill, split, and merge. A user selects those operations interactively to refine a given segmented image. The system observes those actions, and it retains information concerning the regions that were modified. The collected information is stored in a domain knowledge database.

When a user activates the run mode, our system automatically generates a set of rules based on its stored knowledge database. A user, possibly a different person, can load a new image, and the system will automatically apply its rules to update the image. Based on the geometric properties of those regions, the system selects operations and applies them. Ideally, the system will generate a post processed output image that is the same as the original human operator would produce.

A serious problem faced by designers of knowledge-based (“expert”) systems is the knowledge acquisition bottleneck. Traditionally, a knowledge engineer works closely with a domain expert to develop rules and guidelines for a particular application. The rules are encoded, and an inference engine (such as MYCIN-style reasoning) is used to analyze new inputs to the system. Because this process is inherently difficult and time-consuming, active research efforts are in progress within the artificial intelligence community to automate the knowledge acquisition process (e.g., [1, 4, 5, 6, 8, 9]). For our post processing application, we have adopted a variation of supervised learning strategies in which the system can learn through observation of a domain expert. Many different approaches to inductive inference could be adopted in this study, including support vector machines or explanation-based learning. In our current prototype, we have implemented an approach based on the induction of decision trees.

A decision tree is a graph-theoretic tree in which each interior node represents a decision point, conceptually incorporating an IF-THEN-ELSE statement, and each leaf node represents a final class label that should be assigned. A decision tree can be used to encode

knowledge for classifying objects, which are often presented in the form of vectors from a feature space. We have developed a strategy in which computed region properties, along with other image-related properties, comprise the feature space for decision tree induction. As a simple example, consider region size and radial distance from the center of a log slice as two features that can be computed for a given region in an image. It is possible to map any particular region onto a point in this feature space, and to assign a label indicating a desired action, such as “remove this region.”

With such a training set, it is possible to use information-theoretic methods to construct a decision tree that can select an action for all points in the feature space. Among the most common systems for decision-tree induction are ID3, C4.5, CART, and MSM-T [13, 15, 16]. All those decision trees create a classification tree by repeatedly subdividing the feature space, using linear univariate thresholds. The result is a set of separating hyperplanes that are parallel to the feature-space axes, with resulting subsets forming a partition of the feature space. Entropy measures are often used to select which feature variable to be considered at each node of the decision tree.

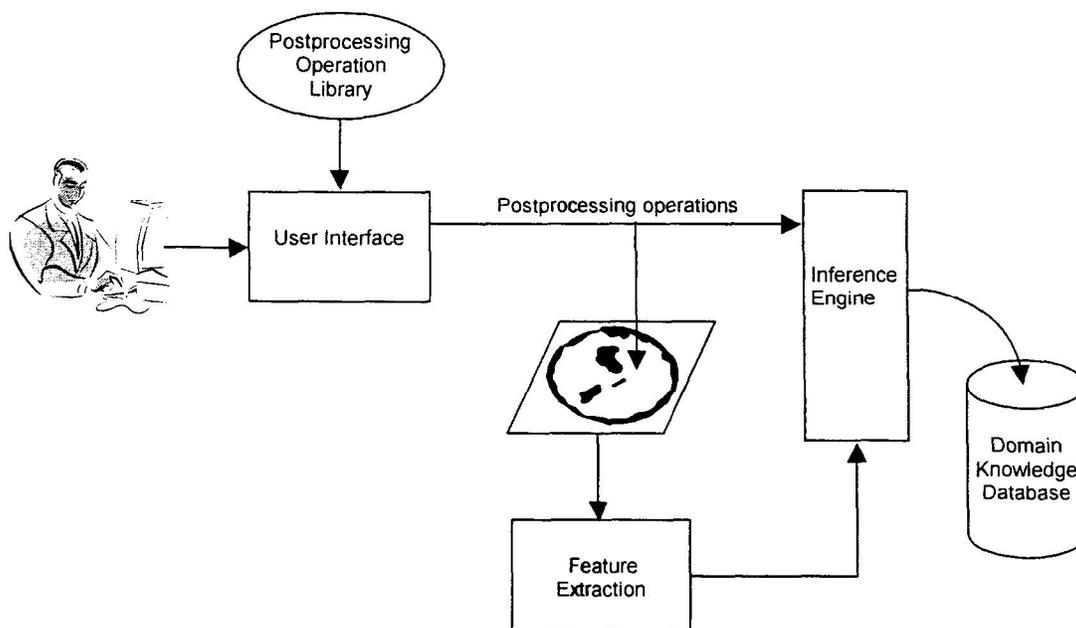


Figure 2. System operation during *learn mode*. A human operator edits a segmented image interactively, as the system observes and extracts information to be used later.

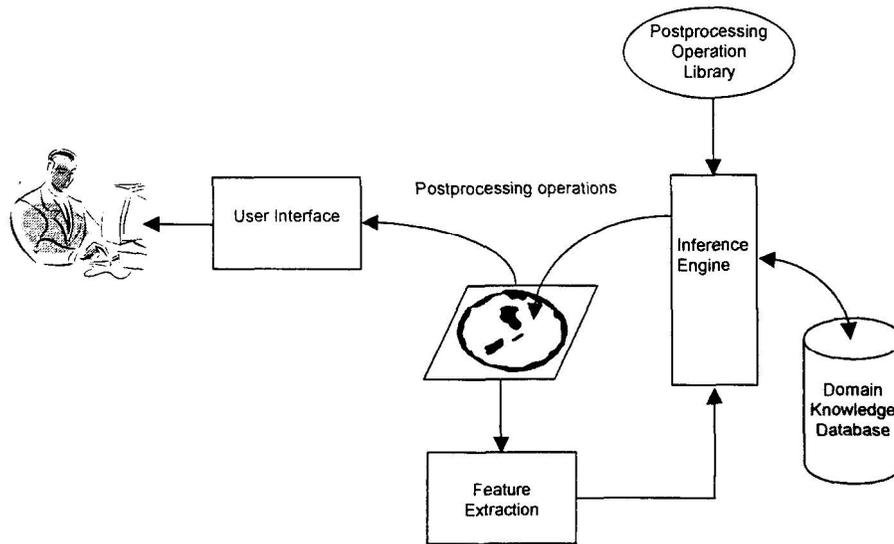


Figure 3. System operation during *run mode*. The user provides segmented images, and the system automatically modifies the image in a manner similar to the human’s earlier editing steps.

Results and Discussion

We have developed a prototype system, which we call *IntelliPost*, that illustrates the fundamental approach for learning postprocessing rules. The user interface, shown in Figure 4, allows the user to select either learn mode or run mode for the session. During learn mode, the user can select regions for modification. For the example results shown here, the only action considered is region removal. The user selects a particular “layer” to modify at any given time, where each layer of the image corresponds to a label assigned by the ANN. For red oak, the layer choices are “knot,” “decay,” “split”, “bark,” and “clear wood.” As the user removes particular regions, the system computes and stores information for each region (such as region size, radial distance, and layer type) in a feature table. When the user later selects run mode: the system automatically calculates and stores feature vectors for the remaining regions in the image: and notes that these are examples of regions that should be retained. The system then constructs a decision tree based on these training vectors. When the user loads a new image to be post processed, the system automatically considers each region in the image, and consults the decision tree to determine which of those regions should be removed and which should be retained.

As an example of system operation, Figure 5 shows a CT slice that was used to train the system. Beginning with the image in part (b) of the figure (previously generated by an ANN classifier), the user removed several regions from the different layers to obtain the result shown in (c). Because the image in (b) contains a total of 177 distinct foreground regions, the system automatically collected 177 feature vectors during the training session. The system then automatically constructed a decision tree using this information.

Using this feature table in run mode, the system post processed several image slices. (These images were not used during the system's training session.) Two of these images are shown in Figure 6: one from the same log (top row), and one from a different red oak log. Column (b) contains images that were tentatively classified by an ANN, and column (c) presents images that were refined by our system automatically. For both image slices, the removal of small regions has improved the quality of the results considerably. In the top row, for example, several small split regions have been correctly removed from the center portion of the image. Other spurious regions, incorrectly labeled split, decay, and clear wood, have similarly been removed from the bark area. Of particular interest is one narrow region incorrectly labeled decay, near the left edge of the log, that has been removed based on the rules encoded in the decision tree. Although it may be difficult to see in the figure, this region is relatively large in area, and would not be removed by a filter based solely on small region size. Our system has learned rules based on both size and distance from the center of the log slice, and that causes this region to be removed in spite of its size. The bottom row of this figure contains a slice from a different log, with large decay region near the center. The ANN has incorrectly labeled several small regions as split and knot, and the postprocessing system has removed them. The postprocessing system has also cleaned up the bark portion of these images considerably.

Figure 7 shows a post processed version of the image from Figure 1. Most of the postprocessing operations were performed automatically and correctly, but for this case an additional unlearned operation was performed. The large bark region near the top of the image, which borders the knot, is actually an extension of the correctly labeled bark region from the perimeter of the log. Hence, region removal is not appropriate in this case. Instead, the system employed a "hard-coded" region-split operation, and we include this example as an illustration of the potential of the system to perform region modifications that are more sophisticated than simple region removal. Work is currently in progress to automate the learning of this operation. as well as others.

These results were obtained using a PC (Pentium 4, 1.13 GHz, 256 MB), and the system is implemented with Matlab (R13, version 6.5). During run mode, post processing required approximately 10 seconds per image. As described earlier; the system is capable of incremental learning, which means that the user does not have to have available all training images initially. This is possible because all training vectors are retained from all previous training sessions. When converting to run mode, a full decision tree is computed from the full set of training vectors.

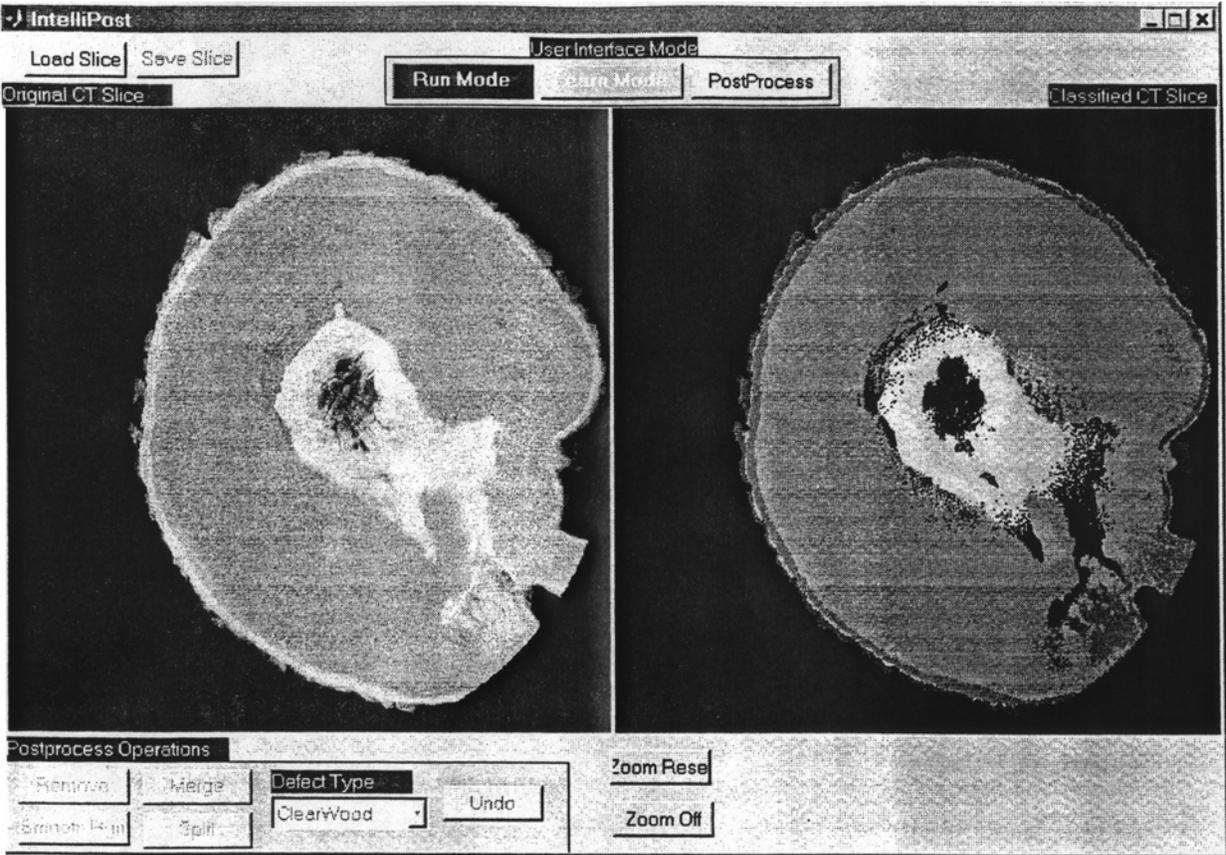


Figure 4. Screen capture of the postprocessing system’s user interface. The input image appears on the left, and the classified image (being modified) is on the right. In learn mode, the user modifies the segmented image interactively. In run mode, as shown above, the system operates automatically.

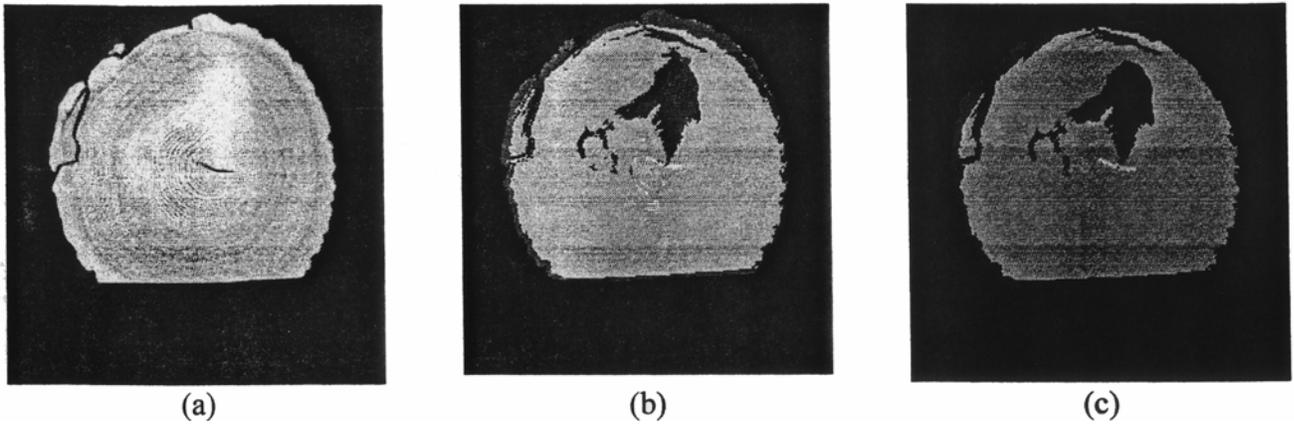


Figure 5. The CT slice from a red oak log used for training. (a) Original CT image. (b) Result from ANN classification. (c) Result of postprocessing, obtained by manually modifying the image in (b).

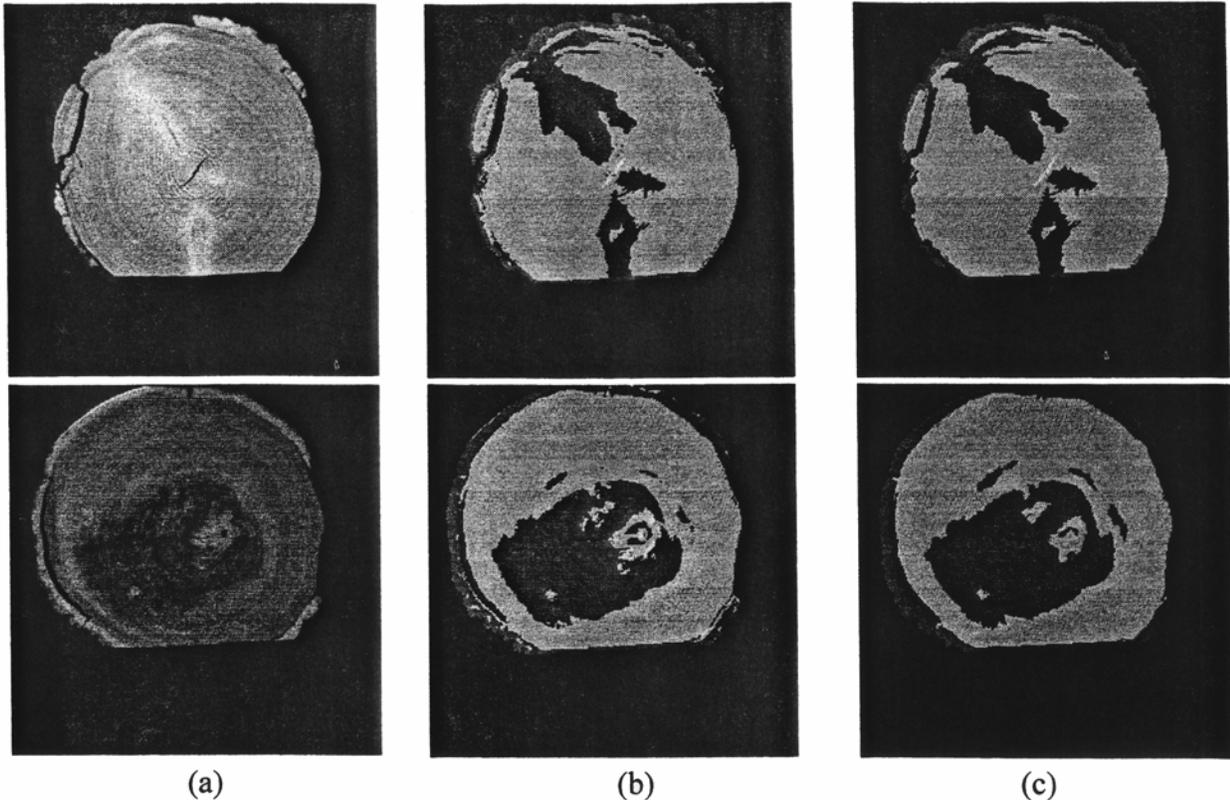


Figure 6. Classification results for two red oak log slices, taken from two different logs. (a) The original CT images. (b) Initial classifications performed by ANN. (c) Automatic postprocessing results, in which many spurious regions have been removed.

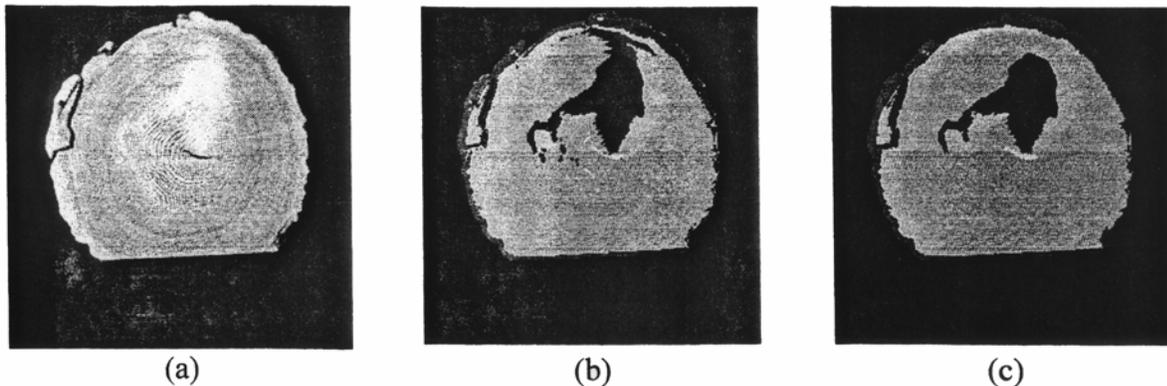


Figure 7. Classification results for the red oak slices from Figure 1. (a) The original CT image. (b) Initial classifications performed by ANN. (c) The result of post-processing. In addition to automatic learning, the system has applied a “hard-coded” operation to remove a portion of the large bark region near the top of the image.

Figure 8 shows preliminary results for two additional species. For these cases, IntelliPost has not been used for post processing. Instead, new ANNs have been trained for each of the species, and a simple majority filter has been applied as the post processing step. These results illustrate the ability of our ANN-based approach to assign reasonable initial labels to a

variety of species. The top row is an example of sugar maple, for which 8 label types were assigned by the ANN. The bottom row is an example of black spruce, and this represents our first attempt to process a softwood species. Only two labels were used by the ANN for this case: “knot” and “clear wood.”

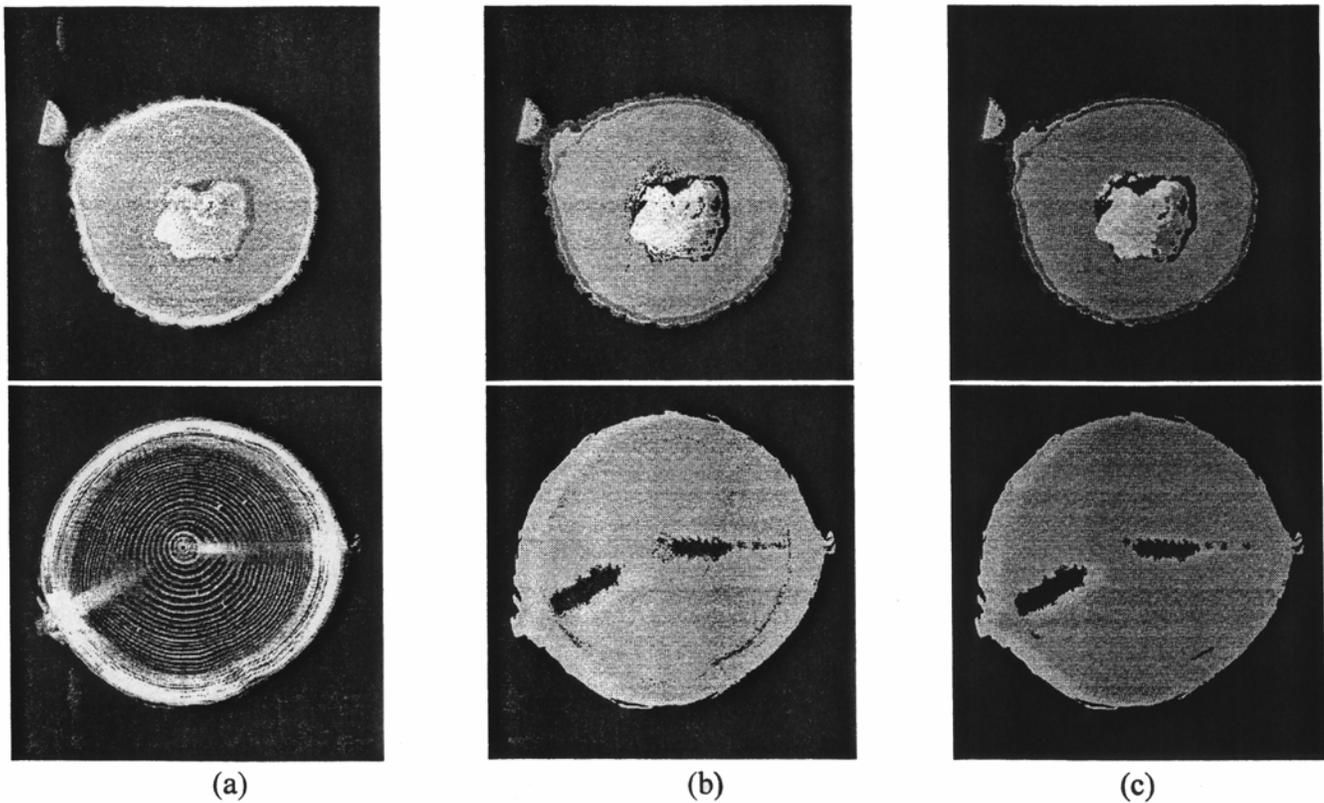


Figure 8. Preliminary classification results for additional species. The top row is sample output for sugar maple, and the bottom row is sample output for black spruce. The post processing results shown here in (c) were not generated by IntelliPost, but instead are the result of a simple majority filter that removes small regions.

Conclusion

This paper has introduced a new approach for refining images in which defects have been detected and labeled. We have developed a prototype system that observes the actions of a human operator who interactively edits a set of test images. The system then applies automated inferencing techniques to develop its own post processing rules based on those actions. After this learning process, the system is capable of automatically applying similar refinement steps to other images.

The system does not simply memorize a sequence of operations by the user, as is often used for robotic teach pendants. Instead, the system develops more general rules based on labeled region properties, such as size, elongation, defect type, and position in the image. Although this approach has been developed particularly for use with CT image slices of

hardwood logs, it is sufficiently general that it can be used for other applications, such as aerial image segmentation.

Acknowledgements

This work was supported in part by USDA National Research Initiative competitive grant #98-35504-6581. We would also like to thank Forintek Canada Corp. for providing CT data sets of sugar maple and black spruce.

References

1. Barstow, D., 1979. *A Knowledge-Based Program Construction*, North-Holland, New York.
2. Bhandarkar, S.M., Faust, T., and Tang, M., 1996. "A system for detection of internal log defects by computer analysis of axial CT images", *Proceedings of IEEE Workshop on Applications of Computer Vision*, Sarasota, FL, pp. 258-263, Dec.
3. Burgess, A.E., 1985. "Potential application of imaging techniques to wood products," *Proceedings of 1st International Conference on Scanning Technology in Sawmilling*, San Francisco, CA, pp. VII 1-13.
4. Fisher, D., 1987. "Knowledge acquisition via conceptual clustering", Technical Report No. 8722, Department of Information and Computer Science, University of California-Irvine.
5. Halbert, D., 1984. "Programming by example", Ph.D. thesis, Department of Electrical Engineering and Computer Science, University of California-Berkeley.
6. Kurlander, D., 1993. "Graphical editing by example", Ph.D. thesis, Department of Computer Science, Columbia University.
7. Li, P., Abbott, A. L., and Schmoldt, D. L., 1996. "Automated analysis of CT images for the inspection of hardwood logs", *Proceedings of the IEEE International Conference on Neural Networks*, Washington, DC, pp. 1744-1749.
8. Lieberman, H., 1992. "Capturing design expertise interactively by example," *Proceedings of the East - West International Conference on Human-Computer Interaction*, ICSTI, St. Petersburg, USSR, pp. 360-366, August.
9. Mulsby, D., Witten, I., Kittlitz, K. and Franceschin, V., 1992. "Inferring graphical procedures: the complete metamouse", *Human-Computer Interaction*, Vol. 7. No. 1, pp. 47-89.
10. Ocoña, L. G., 1991. "Computer integrated manufacturing issues related to the hardwood log sawmill," *Journal of Forest Engineering*, Vol. 3, No. 1, pp. 39-45.
11. Schmoldt, D.L., Li, P., and Abbott, A. L., 1997. "Machine vision using artificial neural networks and 3D pixel neighborhoods," *Computers and Electronics in Agriculture*, Vol. 16, No. 3, pp. 255-271.
12. Schmoldt, D.L., He, J., and Abbott, A.L., 2000. "Automated labeling of log features in CT imagery of multiple hardwood species," *Wood and Fiber Science*, Vol. 32, No. 3, pp. 287-300.
13. Street, W. Nick, "Oblique multicategory decision trees using nonlinear programming", to appear in *INFORMS Journal on Computing*.

14. Tsolakides, J.A, 1969. "A simulation model for log yield study," *Forest Products Journal*, Vol. 19, No. 7, pp. 21-26.
15. Quinlan, J.R., 1993. "C4.5: Programs for machine learning", Morgan-Kaufman, San Manteo, CA...
16. Quinlan, J.R., 1986. "Induction of decision trees", *Machine Learning*, Vol. 1, pp. 81-106.

SCANTECH 2003

**THE TENTH INTERNATIONAL CONFERENCE ON
SCANNING TECHNOLOGY AND PROCESS OPTIMIZATION
IN THE WOOD INDUSTRY**

NOVEMBER 4-5, 2003
HOLIDAY INN SEATTLE INTERNATIONAL AIRPORT
SEATTLE, WASHINGTON, USA

PROCEEDINGS

SPONSORED BY

WOOD MACHINING INSTITUTE

IN COOPERATION WITH

FOREST PRODUCTS SOCIETY

AND

**INTERNATIONAL UNION OF FOREST
RESEARCH ORGANIZATIONS**

Copyright© 2003 by Wood Machining Institute, PO Box 476, Berkeley, CA 94701, U.S.A. All rights reserved.
No part of this publication may be Not be reproduced or copied without the written permission of Wood Machining
Institute, except in the case of quotations embodied in articles and reviews.