COMPUTER INTERPRETATION OF COMPUTED TOMOGRAPHY SCANS OF SAWLOGS

by

Edwin C. Bryant

B.Sc., University of British Columbia, 1978.

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF

THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE

in the Department

of

Computing Science

SIMON FRASER UNIVERSITY

August 1983

APPROVAL

Name: Edwin C. Bryant

Degree: Master of Science

Title of Thesis: Computer Interpretation of Computed Tomography Scans of Sawlogs

Examining Committee:

Chairperson: Wo-Shun Luk

Brian Funt
Senior Supervisor

Thomas Calvert

Nick Cercone

Donald George
External Examiner
Professor
Faculty of Engineering Science
Simon Fraser University

Date Approved: July 21, 1983

PARTIAL COPYRIGHT LICENSE

I hereby grant to Simon Fraser University the right to lend my thesis, project or extended essay (the title of which is shown below) to users of the Simon Fraser University Library, and to make partial or single copies only for such users or in response to a request from the library of any other university, or other educational institution, on its own behalf or for one of its users. I further agree that permission for multiple copying of this work for scholarly purposes may be granted by me or the Dean of Graduate Studies. It is understood that copying or publication of this work for financial gain shall not be allowed without my written permission.

Title of Thesis/~~Project/Extended Essay~~

Computer Interpretation of Computed Tomography Scans of Sawlogs.

Author:

(signature)

Edwin C. Bryant.

(name)

83/08/16.

(date)

Abstract

A Computed Tomography scan of a sawlog produces a cross-sectional density map. Image processing techniques, directed by knowledge about the internal structure of logs, can be applied to segment a CT-scan into regions corresponding to good wood and regions corresponding to defects, such as knots and holes. This thesis describes algorithms necessary to interpret a computed tomography scan of a sawlog. Results are presented to show the algorithm's robust performance on log samples of differing wood species and water content.

Acknowledgements

I would like to acknowledge the support and assistance of my supervisor, Dr. Brian Funt. Stanley Wong implemented the algorithm to perform histogram analysis. Charles Brown and Brian Wideen helped implement the IIS support software. Many of the subroutines to control the IIS were initially obtained from the University of Wisconsin at Madison. Log samples were obtained through the assistance of J. Aune of MacMillan Bloedel Research Inc. CT-scanning of log samples was performed by A. Burgess and H. Ghanderharrian of the Department of Radiology at the University of British Columbia. This research was funded by British Columbia Science Council Research Grant #47 (RC-5).

Table of Contents

List of Figures

List of Figures

1. Introduction

In an era of declining quantity and quality of sawlogs, it is important to mill sawlogs to maximize the yield of usable lumber. Internal defects in the log, such as knots or rot have to be determined and avoided when the sawlog is cut. Computed Tomography scans of logs non-destructively reveal the internal structure of growth rings, knots, and rot.

In this thesis, algorithms capable of interpreting a Computed Tomography scan of a log are described. In particular, they determine the high density regions corresponding to knots, the low density regions corresponding to cracks and holes, and the clear wood and rotten wood in the remaining regions of medium density. These algorithms incorporate knowledge about the internal structure of logs to guide the CT-scan image interpretation system.

It should be noted that the research described in this thesis represents the first step in automating the process of determining cutting strategies for sawlogs. The algorithms presented here interpret a single CT-scan image of a log. Work remains to be done on linking adjacent interpreted CT-scan images into a complete three-dimensional model of the log, which could then be used to determine the optimal cutting strategy.

The organization of this thesis is as follows. First, research related to CT-scan image interpretation by computer will be discussed. After this, a brief introduction to the International Imaging Systems (IIS) image processing system, which was used extensively in this research, will be presented. Next, algorithms to interpret CT-scan images of logs will be presented, including implementation details. Finally, the results of interpreting a wide variety of log samples, including different species, will be shown.

2. Related Research

Computed Tomography scanning is an X-Ray imaging technique primarily used in the field of bio-medical imaging. A CT-scan is a cross-sectional view of an object reconstructed from a number of X-Ray projections. A CT-scan is typically displayed as a digital image, with pixel values corresponding to the X-Ray attenuation of the object at that particular point. To be more precise, a pixel value corresponds to the average attenuation in the volume of the object corresponding to the boundaries of the pixel and the thickness of the CT-scan. CT-scan cross-sections of an object can be as narrow as 2 or 3 mm (Gordon [1975] Katz [1978] Herman [1979]). Since X-Ray attenuation is proportional to density, the output of a CT-scan can be treated as a density map.

In this research the CT-scanning process itself was not of concern. Various factors, such as beam hardening (the upward shift in the average energy level of X-Rays as they pass through an object), the number of projections, and the choice and accuracy of a reconstruction algorithm could, however, have substantial effects on the quality and cost of the scans. The sample logs analysed in this thesis were scanned using a CT-scanner at the University of British Columbia Acute Care Hospital. When the sample logs were scanned, appropriate parameters were chosen by the scanning unit staff which they felt yielded the best results in terms of image resolution and

contrast.

Machine interpretation of medical CT-scans has been studied and there are some working computer interpretation systems. Generally, their goal is to locate abnormalities in CT-scans of human anatomy. Typical results are Hwang [1979], Rhodes [1979], and Selfridge and Prewitt (Selfridge [1981]). Hwang segments single CT-scans into regions to locate abnormalities, as do Selfridge and Prewitt, while Rhodes determines three-dimensional structures from adjacent CT-scans.

Hwang segments CT-scans in two phases. The first phase divides the scan into atomic regions by linking adjacent pixels of similar intensity together. The similarity measure is determined by locating the local maxima of the histogram of all pixel values and setting the similarity boundaries half way between adjacent local maxima. The second phase is to determine larger regions by using the atomic regions to mask the texture locally.

Rhodes implements a simpler technique, on adjacent CT-scans, which links neighbouring pixels with similar density values to form three-dimensional volumes. Special control structures are needed to correctly handle the case where a three-dimensional region grows through a particular CT-scan image and then re-enters it later at a different location.

Selfridge and Prewitt implemented a computer system to "delineate specific organs by incorporating some anatomical knowledge". They describe their algorithm as "regionally iterative, adaptive boundary-delineation for specific structures". In particular, they tailored these programs to the detection of kidneys in abdominal CT-scans. Using the assumption that human organs are of uniform density throughout, they initially select a region in the vicinity of the known location of the kidneys. This region of uniform density must have an area of at least nine pixels. They used region growing from this initial position to extend the boundary of the region by iteratively cycling through all adjacent boundary points. For each two adjacent boundary points, a calculation is performed at the mid-point between them to determine if the midpoint is inside or outside the region of interest. If the midpoint is inside the boundary, then the boundary at that point is moved out. The reverse is true if the mid-point is outside the boundary.

> "Reasonableness of performance can be monitored as the iteration progresses. For example, inappropriate initial points or parameter settings might be detected by subsequently generated implausible changes in area or boundary shape. The algorithm could search for a starting point that leads to stability" (Selfridge [1981] p. 268).

The initial regions and parameters, as well as the adaptive boundary delineation parameters are set by the users of the system. Presumably parameter settings would become fixed over

time as optimal values for human CT-scans are found.

Whether explicitly stated or not, these systems rely on the fact that CT-scans of human anatomy possess certain characteristics that can be exploited. The most obvious of these is the consistency of human anatomy across subjects.

Region growing has also been used in the interpretation of the log scans, although much differently. The boundary of the initial regions is determined by histogram analysis, rather than region uniformity. Further, regions were grown to remove holes, to merge suitable adjacent regions, and, more importantly, to enable their boundaries to be described in a simple manner. As well, the parameters which controlled the region growing are generated automatically.

Although CT-scans have not been applied in a sawmilling environment to date, conventional X-Rays have been applied, at least on an experimental basis since the mid 1960's WWP [1968], Caplan [1967a], Caplan [1967b], and Miller [1964]. X-Rays were used to automate the sawmilling process in the following way. After the bark was removed from a log, it would pass through a sensor station. The station consisted of two television cameras viewing opposite sides of the log, and a conventional X-Ray device. The three images (two surface views and the X-Ray) were presented simultaneously on a group of three monitors. A human operator viewing all three screens as the

log passed through the sensor station would press one of four buttons to indicate the grade of the log. This system was technologically advanced for the mid 1960's, but due to the advances in Computed Tomography and the work on computer image interpretation the process of determining the internal structure of a sawlog can now be totally automated, and the resulting interpretation better utilized.

CT-scanning has recently been applied in the field of non-destructive testing (Ellinger [1979], Hopkins [1981], Reimers [1980], Kruger [1978a], and Szymani [1981]). There is also related research on the development of Computed Tomography devices for industrial applications (Kruger [1978b]). It is hoped that industrial CT-scanners will be less costly than medical scanners, since less projections may be necessary if higher radiation levels are used.

A CT-scan of a log reveals its internal structure in surprising detail. In addition to pointing out the suitability of wood for CT-scanning, Hopkins [1981] also suggests histogram analysis of various regions of the log as a possible interpretation method. This method is inadequate for the complete analysis of the internal structure of a log, since no mention is made of how these regions are to be determined, nor how to compensate for differing log species or water content. The classification of rotten wood is also not discussed.

3. Research Equipment

If this thesis were to be implemented commercially, rapid execution of the CT-scan interpretation program would be a necessity. CT-scanning a thirty foot log every six inches, for example, would generate sixty scans. These would have to be interpreted in only a few seconds.

The execution time of the interpretation algorithms was reduced to approximately 90 seconds by utilizing an International Imaging Systems model 70F interactive digital image processing system (IIS). Almost all of the log interpretation system uses the capabilities and features of the IIS.

The IIS can be logically divided into two functions. The first function displays the data stored in its image channels. The second function performs image processing functions, as controlled by the host computer. The display process will be described first.

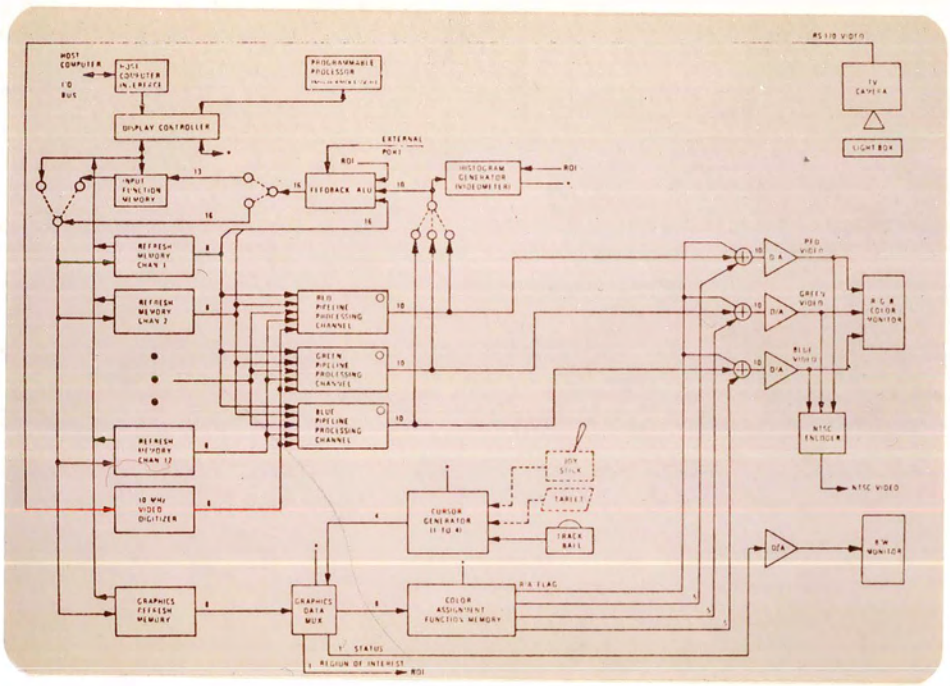


Figure 1.
IIS Design Overview
from IIS [1979]

A schematic of the IIS is shown in Figure 1. The IIS currently consists of four image channels, each one 512 by 512 pixels with 8 bits resolution per pixel. A CT-scan of a log can be stored in one image channel. There are three independent colour pipeline processors which are used to generate the red, green, and blue video signals which in turn drive a colour monitor. Under host computer control, any channel can be displayed through any number of pipelines. To display a CT-scan image stored in a particular channel as a black and white image, that channel is displayed through all three of the colour pipeline processors simultaneously.

Many of the display features of the IIS are table driven. This is a very flexible and rapid way of mapping a given set of input values to a set of output values. An example of this is the lookup table (LUT) which takes as input pixels from an image channel. The LUT maps the input values to output values and passes them to the pipeline colour processors. There are three LUTs for each channel, one for each of the three pipeline colour processors. In the default case, these LUTs perform an indentity mapping between input and output pixel values.

There is also a special-purpose graphics channel, consisting of four 512 by 512 bitplanes. They can be used to display text on the colour monitor. More importantly, the 1's in a particular bitplane can be used to form a region of interest (ROI). ROI's are used for image processing by controlling the feedback arithmetic/logic unit, described later.

Each pipeline colour processor takes as input the sum of all the channels displayed through it. A pipeline processor can scroll and zoom any channel displayed through it. The zoom function is implemented in hardware and magnifies the image channel by a factor of 2, 4, or 8. Image scrolling is simply a re-definition of the origin (the upper left corner) of the display. The IIS allows images to wrap around the display. For example, the parts of an image scrolled off the bottom of the display would reappear at the top of the display. After

the image channels displayed through a pipeline colour processor have been scrolled, zoomed, and summed, they are mapped through another table, called the output function memory (OFM). OFMs function in the same manner as LUTs. The output of the OFM for each of the pipeline colour processors is fed through a D/A converter, which generates the red, green, and blue video signals which drive the monitor.

The second function of the IIS is to process images stored in its image channels. This processing can be divided into two groups: that done at image display time, such as image inversion (negation), false or random colouring, and contrast balancing; and that requiring image feedback and accumulation through the arithmetic/logic unit, such as image convolution and region growing.

Simple image processing, which can be done at image display time, is implemented by modifying the lookup tables associated with a displayed channel or the output function memory of a pipeline colour processor. For example, to perform image inversion, the appropriate LUT is loaded such that the lowest pixel value (zero) is mapped to the highest value (255), and the highest to the lowest. Although the image data has not been altered, an inverted image will be displayed. Note that this image inversion is re-computed every 1/30th of a second. In contrast, more complex image processing typically requires the change of the original image channel data, or the

accumulation of altered image data in another channel by using the feedback arithmetic/logic unit.

The IIS has an arithmetic/logic unit (ALU) which can process all of the pixels in an image in one video frame time (1/30th of a second). The ALU has 2 image inputs, the output of one of the pipeline colour processors, and the accumulator. The choice of which pipeline colour processor output to use as input to the ALU is under user control. The accumulator is the unprocessed (by the display hardware) contents of image channel 1, or alternatively channel 1 and 2 treated as a 16 bit quantity. The input and ALU output data can have 16 bits of accuracy. When a 16 bit accumulator is used, image channel 2 contains the 8 most significant bits of the data, and image channel 1 the 8 least significant bits. The two image channels are passed together to the ALU as a single 16 bit quantity. The ALU can be loaded with any of over 40 functions. The input and output of the ALU can be further modified, at the discretion of the user, to scale or sign extend the data. As mentioned, a region of interest (ROI) bitplane from the graphics channel can be used to alter the performance of the ALU. In this case, two ALU functions are loaded, one function to be performed when the ROI bitplane pixel contains a 1, and another function to perform when the ROI bitplane pixel contains a 0. ROIs are used in the interpretation of CT-scans to select the output of an image-processing step (such as

convolution) only in specific regions (such as low density). There is another table associated with the ALU, called the output select table (OST), which allows the user to select different outputs from the ALU depending on the contents of the ROI bitplane, and ALU status flags (such as carry overflow on addition). These IIS features are used to perform image convolutions and other complex image processing functions.

Image convolution is the application of a two-dimensional moving filter to an image. As an example of digital image convolution, consider the following 3 by 3 mask:

0.4	0.4	0.4
0.4	0.8	0.4
0.4	0.4	0.4

This image smoothing mask works in the following way. A pixel in the convolved image is 0.8 times its value in the original image, plus 0.4 times the pixel values of each of its 8 adjacent neighbours. The range of pixel values in the resulting image is four times as large as that in the initial image. If all the pixel values in the initial image under the mask are 255 then the pixel value in the resulting image is:

$$0.8 * 255 + 8 * (0.4 * 255) = 1020 \qquad (1)$$

The convolution can be implemented using the IIS as follows:

- Assume the image is in channel 3.
- The result of the convolution will be accumulated in channels 1 and 2 (recall that a channel can hold 8 bits of data, but the maximum result here may be 10 bits).
- Clear channels 1 and 2.
- Load the alu with the add function.
- For each value in the mask:
  - Write an output function memory (OFM) table that has a slope equal to the mask value. That is:
    OFM_output_value = channel_input_value * mask_value.
  - Scroll the input image (channel 3) so that the pixel corresponding to the mask value is aligned with the central pixel.
  - Feed the image back through the alu, adding in the current contents of channels 1 and 2, and store the result back into channels 1 and 2.
- At this point, channels 1 and 2 hold the resulting smoothed image. We need to scale this image to 8 bits accuracy.
- Write LUT tables that scale the data in channels 1 and 2 to the correct range. That is:
  LUT_output_value = input_channel_values / 4.
- Feed the output of these LUTs back through a linear mapping OFM and store the result in channel 3.
- Channel 3 now contains the smoothed image.

This image smoothing is accomplished in only 11 feedback operations (each feedback completing in 1/30 second): 1 to clear channels 1 and 2, 9 to perform the convolution, and 1 more to store the resulting image in channel 3. If this two-dimensional smoothing operation were performed on an array of data values in a conventional computer, it would be necessary to perform 2,359,296 floating point multiplications and 262,144 floating point divisions. Any image processing operation that can be specified as an image convolution can be implemented using the above method, and executed very rapidly on the IIS.

A VAX 11/750 is the host computer, providing storage for image files and interpretation programs. There are a group of software subroutines, as well as a large number of programs, which provide the interface between the user and the IIS. This software is supported under the UNIX operating system environment.

4. Algorithms for CT Scan Interpretation

CT-scans of logs exhibit readily discernible features, even to the untrained observer.

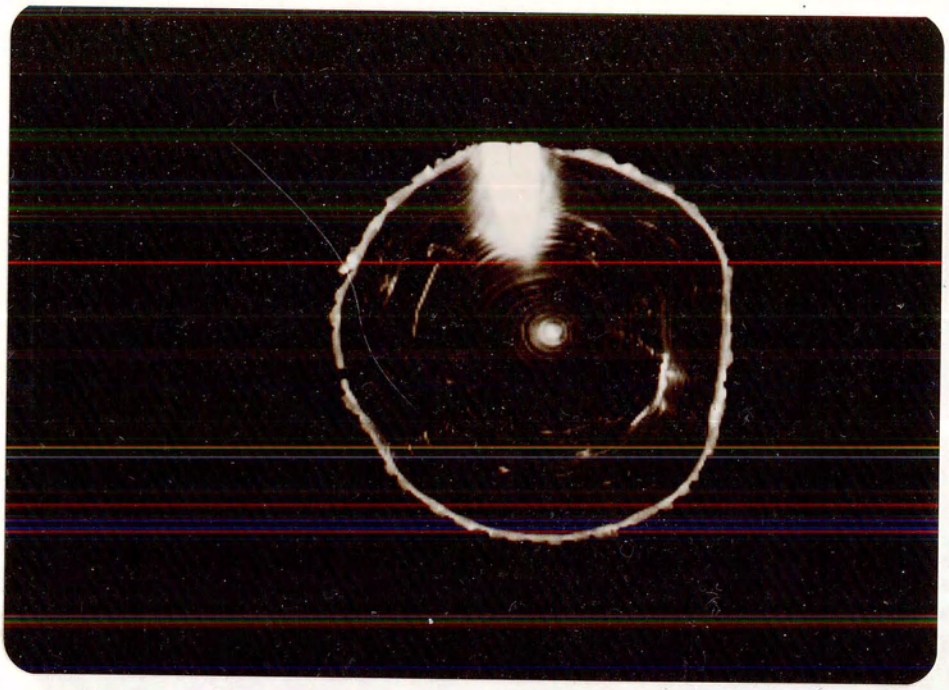


Figure 2.
Sample Log CT Scan.
(log sample of aged hemlock)
The CT-scan is displayed as a black and white image. Higher density areas, such as knots, are displayed as brighter regions. This log sample contains one large knot, located at the top of the scan.

Figure 2 is a sample CT-scan of aged wood; a tree was cut down and let dry for a considerable time prior to obtaining a sample section for CT-scanning. The growth ring structure is immediately evident in this CT-scan. The growth rings are concentric about the center of growth and their variation in intensity is caused by the density difference between winter and summer growth. A knot is also evident in this CT-scan, as

a high intensity, elliptical area. It is a simple task for a naive human viewer to differentiate between good wood and knots. CT-scans of green wood (wood which was only recently cut down) present a more difficult interpretation task.

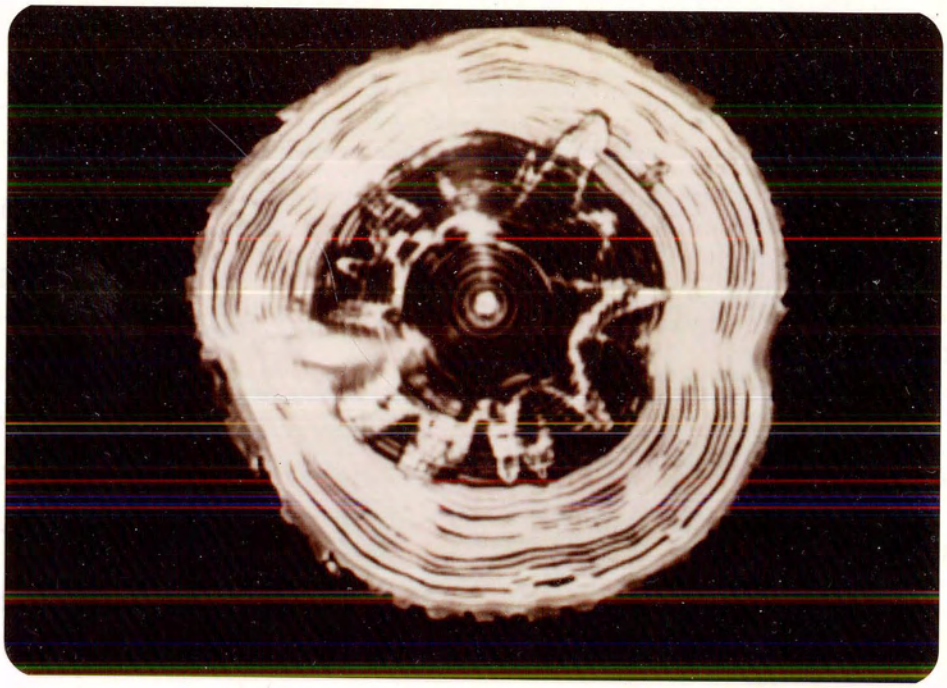


Figure 3.
Sample CT Log Scan.
(log sample of green hemlock)
The CT-scan is displayed as a black and white image. Higher density areas, such as knots, are displayed as brighter regions. This log sample contains a small knot, located at the left of the scan. Higher density sapwood is displayed as a brighter band surrounding the less dense heartwood.

Figure 3 is an sample CT-scan of green wood. This log sample was CT-scanned soon after it was cut down. Both of the log samples shown in Figure 2 and Figure 3 are from the same hemlock tree. CT-scans and interpretation of log samples of different species will be presented in a later section. The

striking difference between the two CT-scans of samples from the same hemlock tree is due to the two types of wood in a living tree: heartwood and sapwood. Sapwood is the outer layer of growth in a tree. As its name implies, sapwood contains more sap than heartwood. Since tree sap in green wood has a very high water content, CT-scans of green wood, as shown in Figure 3, show a brighter band of higher density wood. As a log sample dries, the water content, and hence the density, of the sapwood decreases, until the density approaches the density of the heartwood. The algorithms can account for the possible density difference between heartwood and sapwood, and compensate accordingly. Thus, the amount of time between when a log is cut and CT-scanned does not affect the program's performance.

The computer program segments a CT-scan of a log into regions of four types. These are good wood, rotten wood, knots, and holes and cracks. Good wood is free of knots and rot, and is indicated in CT-scans by uniform concentric growth rings about the center of growth. Holes, cracks, and other low density regions are visible in CT-scans as uniformly dark areas. Knots are high density elliptically-shaped regions, and are visible as brighter areas in CT-scans. The major axis of this elliptically shaped region is always oriented toward the center of growth. Rotten wood may be of higher, lower, or varied density, depending upon water content, and appear in

CT-scans as lighter, darker, or mottled regions, respectively. There are also no growth rings evident in regions of rotten wood. These real-world features, in part, are used to guide the interpretation process.

The CT-scan interpretation process consists of a number of distinct steps. The first is histogram analysis, which results in three pixel values. They grossly divide the image into four types of regions. They are, from lowest the highest density: (i) low density areas, corresponding to the air surrounding the log and internal holes and cracks; (ii) lower density areas corresponding mainly to low density good wood, but possibly including some rotten wood; (iii) medium density areas, corresponding to good wood; and (iv) the high density remainder of the CT-scan, corresponding to possible knots. Further processing of all but the first type of region is required to differentiate between good wood, knots, and rot.

4.1. Histogram Analysis

Histogram analysis determines the threshold values used to segment the CT-scan image into the four basic types of regions.

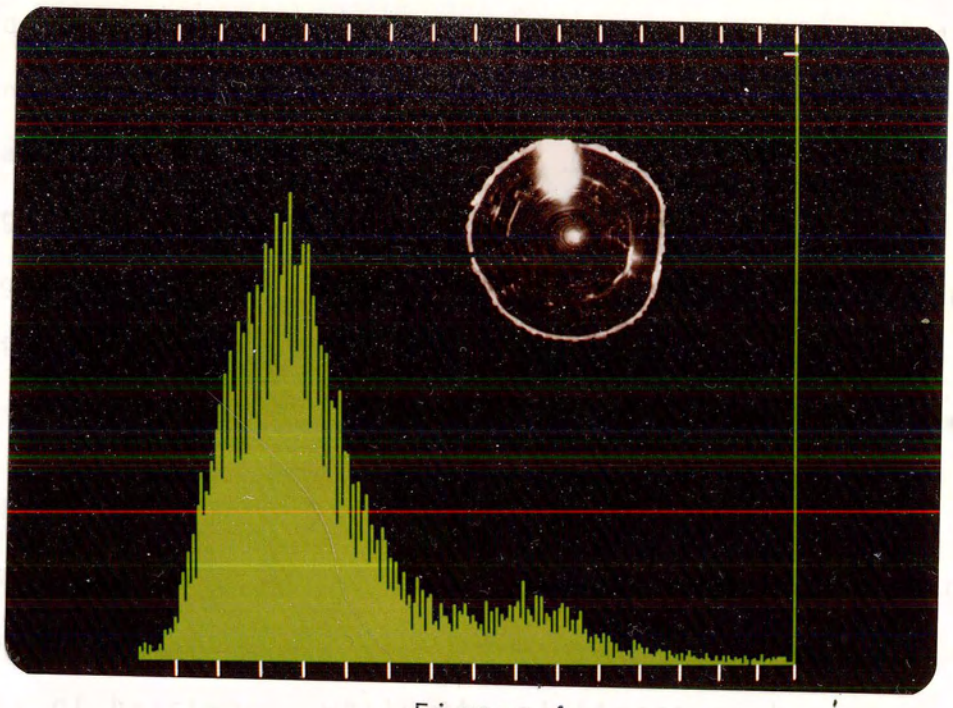


Figure 4.
Sample Histogram
(log sample of aged hemlock)
The histogram has a peak corresponding to the heartwood, and a very pronounced high density peak corresponding to the knot.

A typical histogram is shown in Figure 4. A histogram is a bar graph showing, for each pixel value, the number of pixels with that intensity. A histogram of a CT-scan of a log may have as many as four distinct peaks. The pixel values under the leftmost peak correspond to the density of the air surrounding the log, and holes and cracks internal to the log. The height of this peak is a function of log size and is not important to the analysis. In all the histograms, including Figure 4, the histogram value for the zero pixel value (a relatively large value) is not included since pixels with this value are clearly not dense enough to be wood. Because this value is not

included, the leftmost peak is not present in Figure 4. Its presence or absence does not affect the histogram analysis. At the far right of the histogram is a peak which indicates the average density of the knot regions. This peak in the histogram may not be present in some CT-scans if the log sample contains no knots. Again, the histogram analysis can accommodate the presence or absence of this high density peak.

The middle area of the histogram corresponds to good wood, but may also contain some rotten wood. This region contains one or two peaks. The leftmost peak corresponds to the average density of heartwood, while the rightmost peak corresponds to the average density of sapwood. As a log sample dries, the average density of the sapwood decreases, eventually causing the peaks to merge. Thus, depending on the age of the log sample, the rightmost portion of this middle area of the histogram may contain a distinct peak or may contain a smooth plateau of the right shoulder of the left peak.

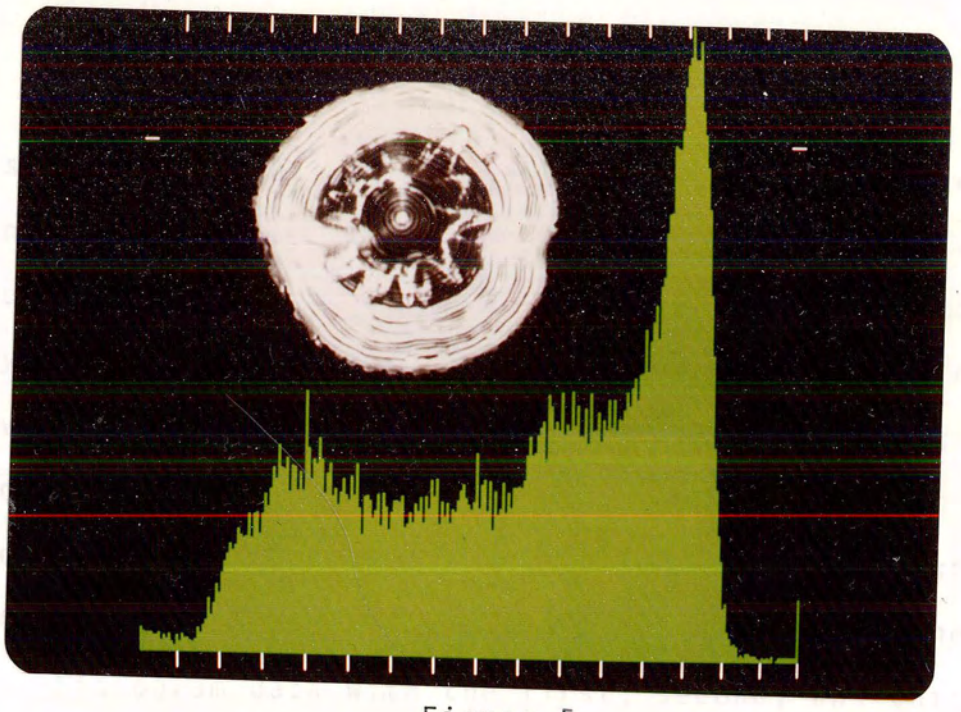


Figure 5.
Sample Histogram
(log sample of green hemlock)
The histogram has three peaks. The sapwood occupies a larger area than the heartwood. There is a small peak at the far right corresponding to the knot.

Figure 5 is a histogram from a sample of green wood. The right peak, corresponding to the average density of sapwood is higher than the peak to the left of it, corresponding to the average density of heartwood.

Histogram analysis proceeds by finding the zero crossings of the second and third derivatives of the histogram, subject to certain constraints. This analysis was influenced by the work of Marr and Hildreth (Marr [1978]), in the following way. Marr and Hildreth proposed a method of detecting edges (all points of local absolute maximum slope) in images by determining the zero crossings of the second derivative of an

image. Here we wish to find the points of maximal change of slope, or curvature. Hence, we take the third derivative of the histogram and locate its zero-crossings. These zero-crossing points correspond to points of maximum histogram curvature, which are generally the maxima (peaks) and minima (valleys) of the histogram. Since we seek only points of positive curvature (the valleys) in the original histogram, we only consider zero crossings of the third derivative at points with non-zero second derivatives. The first, second, and third derivatives of the histogram are found by convolving the original histogram data with the first, second, and third derivatives of a smoothing Gaussian function. The Gaussian function is 50 pixels wide, with a sigma value of 8.

The lower boundary between air and wood is easily found in the histogram using this method. It is the leftmost zero-crossing of the third derivative of the histogram. The upper boundary between possible good wood and possible knots is slightly more difficult to find. This is due to the possibility of there being no knots in the log sample. If there are no knots, the upper region of the original histogram will be relatively flat and near zero in value, producing no discernible peak for knots.

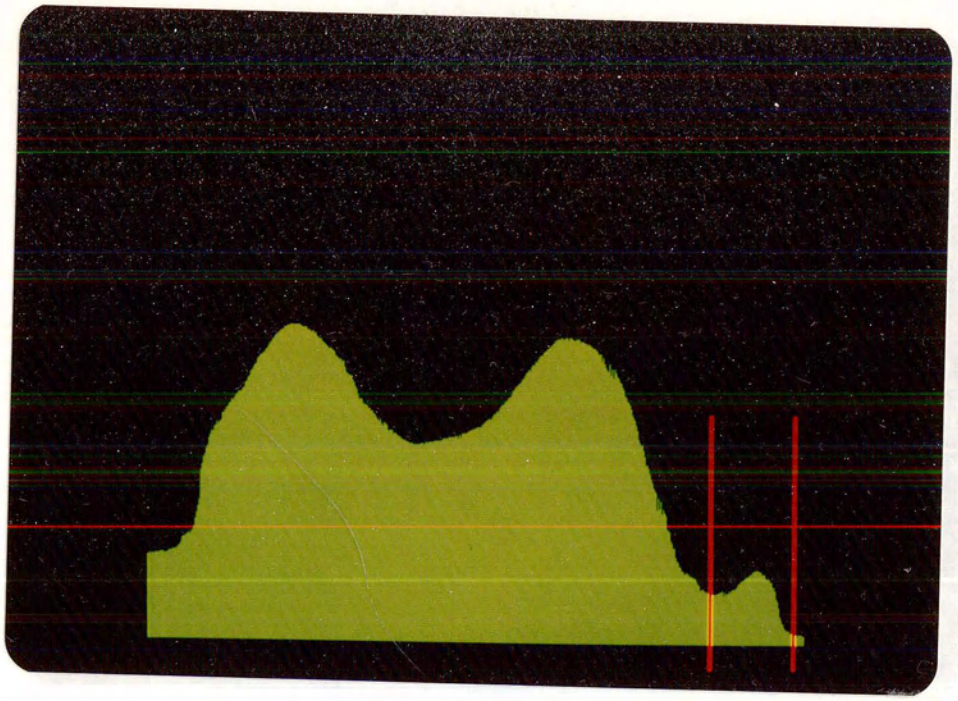


Figure 6.
Typical Histogram with a Knot Peak
This histogram has zero-crossings of the third derivative, with positive curvature, on either side of the knot peak. The lesser of these values would be used as the boundary value between good wood and knots.

In resolving this difficulty, there are two cases to consider. Assume there is a small peak corresponding to the average density of knots, as shown in Figure 6. There are two zero-crossings at the far right of the third derivative of the histogram with non-zero second derivative, one on each side of this peak. These pixel values are used to determine possible knot regions. That is, regions of pixels with values within the above range are possibly knots. These two values are indicated by vertical bars in Figure 6.

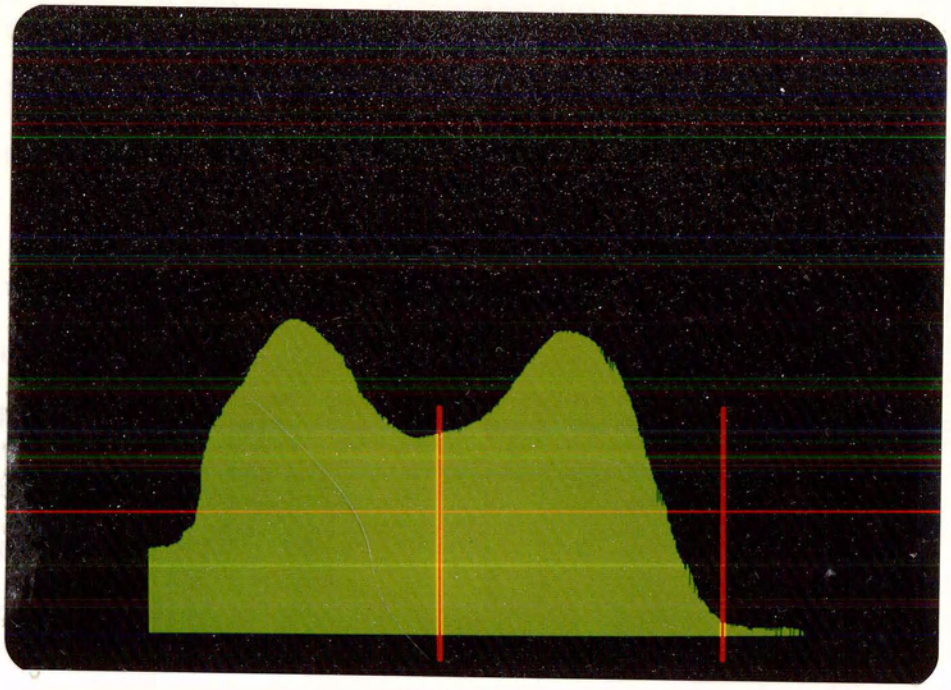


Figure 7.
Typical Histogram with no Knot Peak
Since there is no knot peak, the zero-crossing of the third derivative between the peaks for heartwood and sapwood would be discarded, and the upper zero-crossing would be used.

In Figure 7, there is no discernible peak for possible knots, and the two zero-crossings found by the above method will be the upper boundary of the histogram region corresponding to good wood, and some other zero-crossing point at a much lower density, as shown in the figure. This second point is typically the boundary between heartwood and sapwood, which corresponds to a density value much lower than that possible for knots. Thus, after finding the two uppermost zero-crossings of the third derivative with non-zero second derivative, if the lesser value of these two is well below the possible density of knots, then the greater of these two values

is used as the threshold value to differentiate good wood from possible knots. If the two pixel values of the zero-crossings are both greater than the possible density of knots, then the lesser of these two (corresponding to the left edge of the knot peak) is used as the threshold value to differentiate good wood from possible knots. In the case of the log sample of aged hemlock, the density value for the knot region is so high that the right shoulder of the knot peak is not present. Fortunately, this case is analytically identical to the case when no knot peak is present.

The remaining density value to be determined from the histogram is the boundary between the low density good wood which may contain some rotten wood, and completely good wood. As rot progresses, it causes a breakdown and decay of the internal structure of the log. If the internal decay of the log has not progressed to a point where a hole is created, then the breakdown has at least caused a reduction in wood density to below that of the sound heartwood. A density value is located that is greater than that of the air external to the log, but less that the peak value for heartwood. It is used to threshold the region of good wood which may contain some rotten wood. The upper boundary of the low density good wood, possibly containing some rotten wood, is calculated as the density at the first zero-crossing of the second derivative of the histogram greater than the density previously calculated as

the boundary between air and wood.

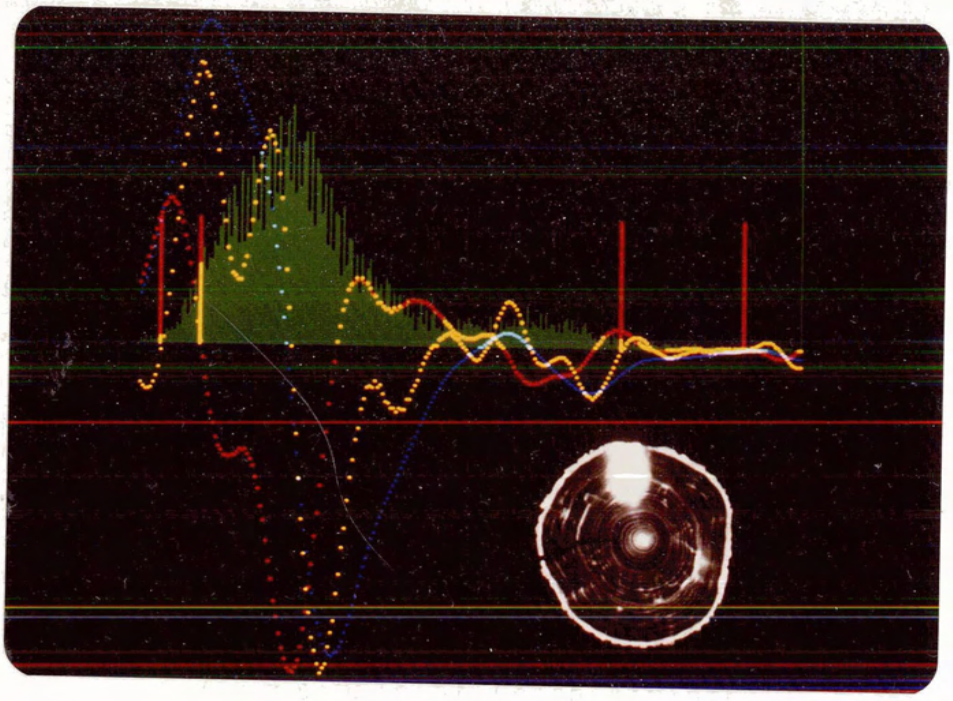


Figure 8.
Results of Histogram Analysis
(log sample of aged hemlock)
The original histogram is shown in green. The first, second, and third derivatives of the histogram are shown in blue, red, and yellow, respectively. The four boundary points determined by histogram analysis are shown in red.

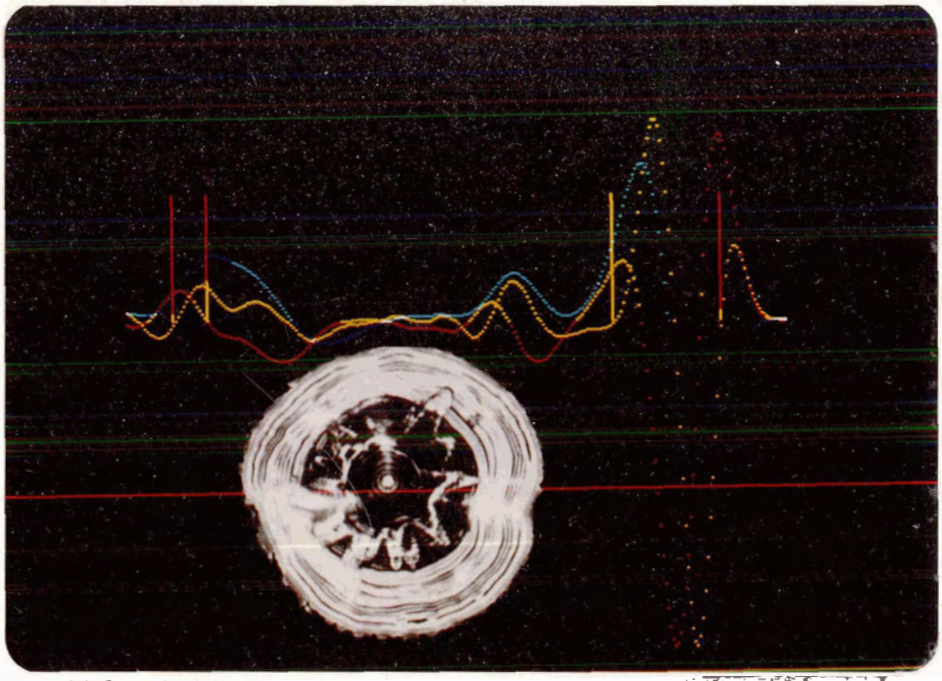


Figure 9.
Results of Histogram Analysis
(log sample of green hemlock)
The original histogram is shown in green. The first, second, and third derivatives of the histogram are shown in blue, red, and yellow, respectively. The four boundary points determined by histogram analysis are shown in red.

Figures 8 and Figure 9 show the results of histogram analysis. The original histogram is shown in green. The first, second, and third derivatives are show in blue, red, and yellow, respectively. The pixel values that result from the calculations described in the text are show as vertical red bars. In both Figure 8 and Figure 9 the lesser of the two upper values is discarded.

4.2. Possible Knot Analysis

High density regions, possibly corresponding to knots, are further processed to determine the likelihood that they are knots. Knots are roughly elliptical in shape, and generally have their major axis pointing towards the center of growth of the log. The center of growth of a log is the center of the concentric growth rings. In most log samples, the center of growth is near the geometric center of the log. Following Baumgart [1974], the major and minor axes of inertia, and the moments of inertia are determined for each region corresponding to a possible knot. These values, combined with other measures, are used to calculate an overall measure of the possibility of a high density region being a knot. These calculations cannot be done using the IIS, and must therefore be implemented using the VAX 11/750 associated with the IIS. Since digital image processing can be very slow on a conventional computer, a very efficient representation of these possible knot regions was needed. An algorithm, described in the next section, was developed and implemented which approximates arbitrary regions by convex regions with straight sides in regular directions.

4.2.1. Region Growing for Convex Shape

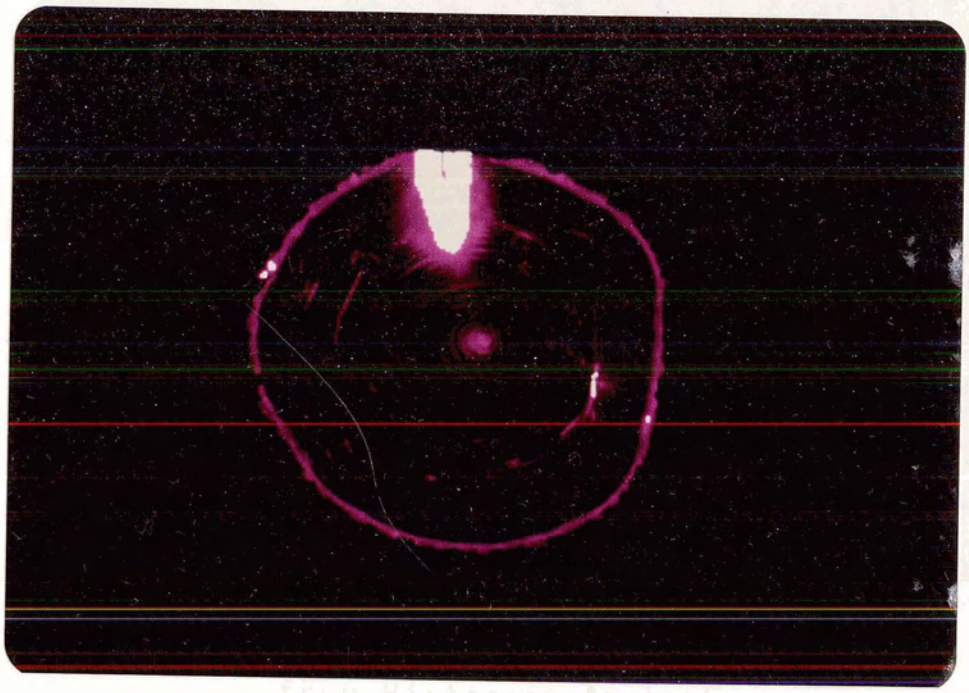


Figure 10.
Possible Knot Regions
from Histogram Analysis
(log sample of aged hemlock)
The original CT-scan is displayed in magenta. The high density areas, which are possibly knots, are displayed in white.

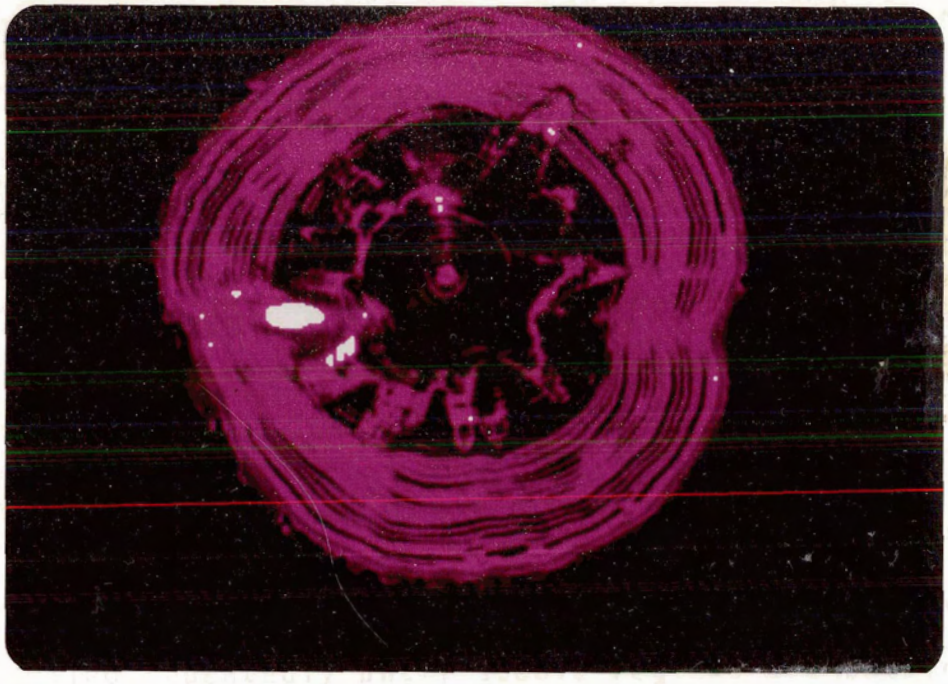


Figure 11.
Possible Knot Regions
from Histogram Analysis
(log sample of green hemlock)
The original CT-scan is displayed in magenta. The high density areas, which are possibly knots, are displayed in white.

Figures 10 and 11 show the binary regions corresponding to possible knots as determined by the histogram analysis. A binary image was formed using the density threshold. A pixel is set to one (on) if the corresponding pixel in the CT-scan has a value in the possible knot range, and set to zero (off) elsewhere. When binary image regions are generated by thresholding, as in this case, the boundaries and extent of the regions are very ill-defined. In an effort to simplify the regions, the region boundaries are smoothed and adjacent regions are joined. Since knots are roughly elliptical in shape, their shape can be approximated by convex regions. An

iterative, locally parallel operation has been devised that grows/prunes arbitrary regions into regular convex regions.

These regular convex regions are defined as follows. Each region is convex and has 8 sides; however, the length of any side may be zero. The direction of each side is limited to one of 8 directions (North, Northeast, East, ...).

This operator is locally parallel since the output at each pixel only depends on its initial value, and the initial values of its 8 adjacent neighbours. This operator is iterative since it is applied repeatedly until stable regions are obtained. The application of this operator to an image of stable regions is equivalent to the identity transform. Stable regions are always guaranteed to result after some number of applications of the operator.

The 8-connected neighbours of a pixel are those eight adjacent pixels in the eight directions (North, Northeast, East, ...). The presence of the 8-connected neighbours to a pixel can be encoded in 8 bits. Here is a possible encoding scheme.

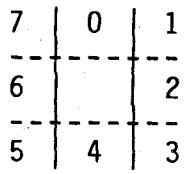
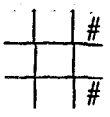


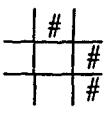
Figure 12.
Encoding Scheme for the 8-Connected
Neighbours of a Pixel

Bits 0 to 7 are set to one if the neighbour in that position relative to the central pixel is present. Otherwise, the bit is set to zero. Since the values of these eight bits are one or zero, there are 256 combinations of possible pixel patterns. If we include the central pixel, with a value of one or zero, there are 512 possible pixel patterns. However, there are many redundant patterns in these 512. If we discard patterns which are identical to another under a rotation by a multiple of 90 degrees, reflections of another pattern around a diagonal (NE-SW or NW-SE), or some combination of both, we reduce the 512 possible patterns to 102.

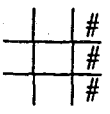


(10)

(11)

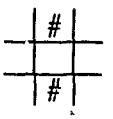


(13)

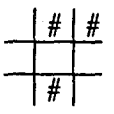


(14)

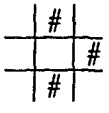
(15)



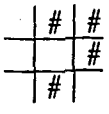
(17)



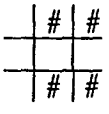
(19)



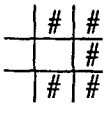
(21)



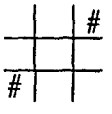
(23)



(27)



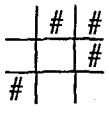
(31)



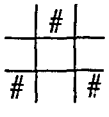
(34)

(35)

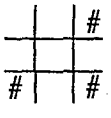
(37)



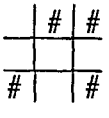
(39)



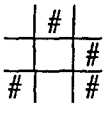
(41)



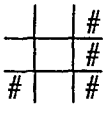
(42)



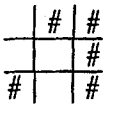
(43)



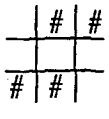
(45)



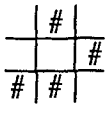
(46)



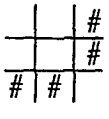
(47)



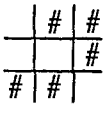
(51)



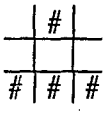
(53)



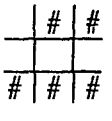
(54)



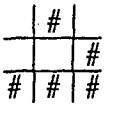
(55)



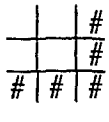
(57)



(59)

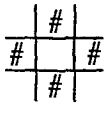


(61)

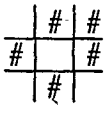


(62)

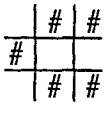
(63)



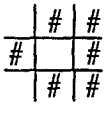
(85)



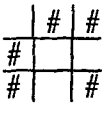
(87)



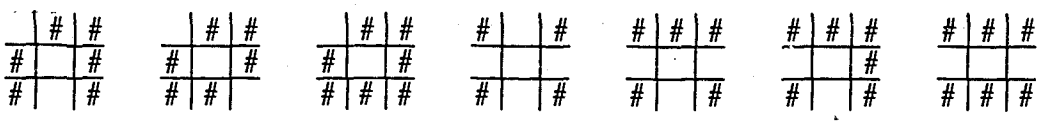
(91)



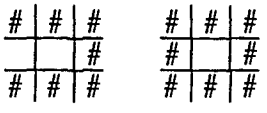
(95)



(107)



(111) (119) (127) (170) (171) (175) (187)



(191) (255)

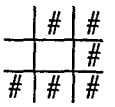
Figure 13.
51 Unique 8-connected Neighbour
Pixel Patterns.
(central pixel not shown)

Figure 13 shows the 51 unique 8-connected neighbour patterns determined from the 256 possible. The number under each pattern is the numeric value assigned to it using the encoding of Figure 12. The center pixel value has not been shown in Figure 13. For each of these patterns, we decide whether to turn the central pixel off, turn it on, or leave it unchanged.

This operator has been implemented on the IIS. An OFM is used to scale the binary data to the appropriate power of 2 necessary to accumulate the data in a given bit position. Using the horizontal and vertical scroll capabilities of the binary data channel, in conjunction with the proper OFM, the binary data can be fed back successively into an accumulator channel. Thus, after eight feedback operations the accumulator channel contains the necessary coding of the 8 connected

neighbours of each pixel in the original binary data. This neighbour encoded data, together with the original binary data is then summed, passed through another pre-computed OFM, which decides whether the pixel will be zero, one, or unchanged, and written over the original binary data channel.

For example, consider the pattern:



(63)

If the central pixel is 0 (a blank), then this pattern represents a concavity in the boundary of the region, and the central pixel will be made a 1 (a #). If the central pixel is a 1 (a #), then this pixel will remain unchanged. Thus, if pixel pattern 63 represents the location of the 8-connected neighbours of the central pixel, then the central pixel will always be turned on.

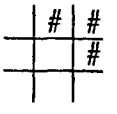
As another example, consider the pattern:

(1)

If the central pixel is 0 (a blank), then this pattern

represents a pixel next to a solitary pixel, or next to end of a group of pixels, and should remain 0 (a blank). If the central pixel is 1 (a #), then this pattern represents a pixel at the end of a group of pixels, and should be made a 0 (a blank). Thus, if pixel pattern 1 represents the location of the 8-connected neighbours of the central pixel, then the central pixel will always be turned off.

As a last example, consider the pattern:



(7)

If the central pixel is 0 (a blank), then this pattern represents a pixel adjacent to a diagonal boundary of a region, and the central pixel will be unchanged (a blank). If the central pixel is 1 (a #), then this pattern represents a pixel at the corner of the region, at the junction of horizontal and vertical boundaries of the region, and the central pixel will be unchanged (a #). Thus, if pixel pattern 7 represents the location of the 8-connected neighbours of the central pixel, then the central pixel will always be unchanged.

At each application of the operator, the new binary image output by this operator is used as input for the next application of the operator. This operator can be applied to a

binary image approximately three times per second.

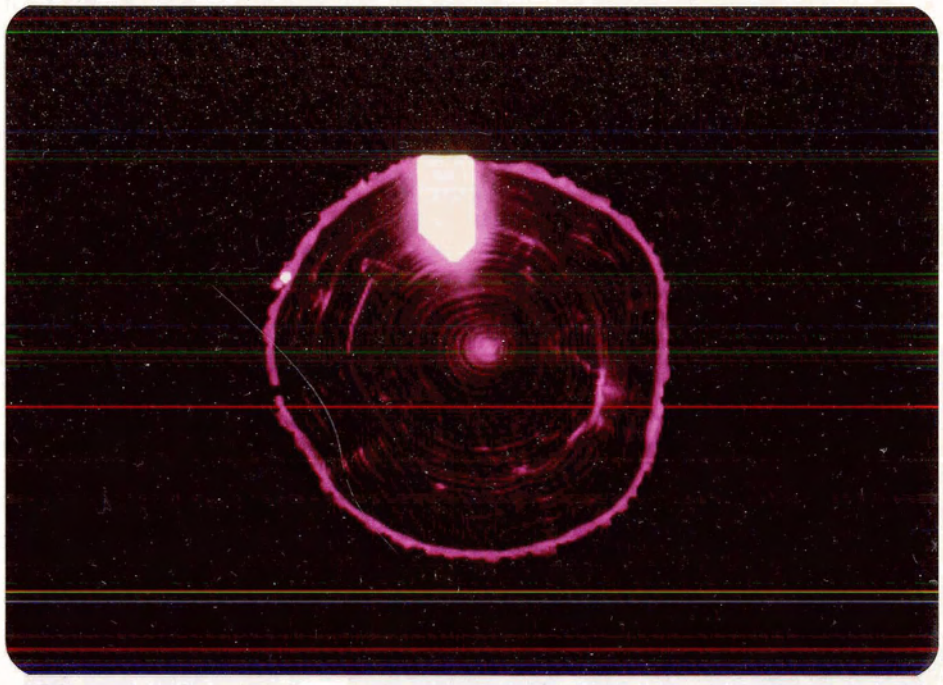


Figure 14.
Possible Knot Regions
after Region Growing
(log sample of aged hemlock)
The original CT-scan is displayed in magenta. The high density areas, now with uniform shape, are displayed in white.

This operator, although quite simple to specify and implement, has some very powerful properties. The most important is that a stable image will always result after a finite number of applications of the operator. That is, after some number of applications of the operator, successive applications will not change the resulting image. The number of iterations necessary to develop a stable image is proportional to the maximum distance from the boundary of the initial region to the boundary of the eventual convex region.

The initial high density regions may completely surround a low density area, forming a hole. These holes are filled in as the iteration continues. If the new uniform convex boundaries of some initially distinct regions overlap, then these initial regions are merged into a single uniform region. Single pixels, or rows of pixels, are pruned.

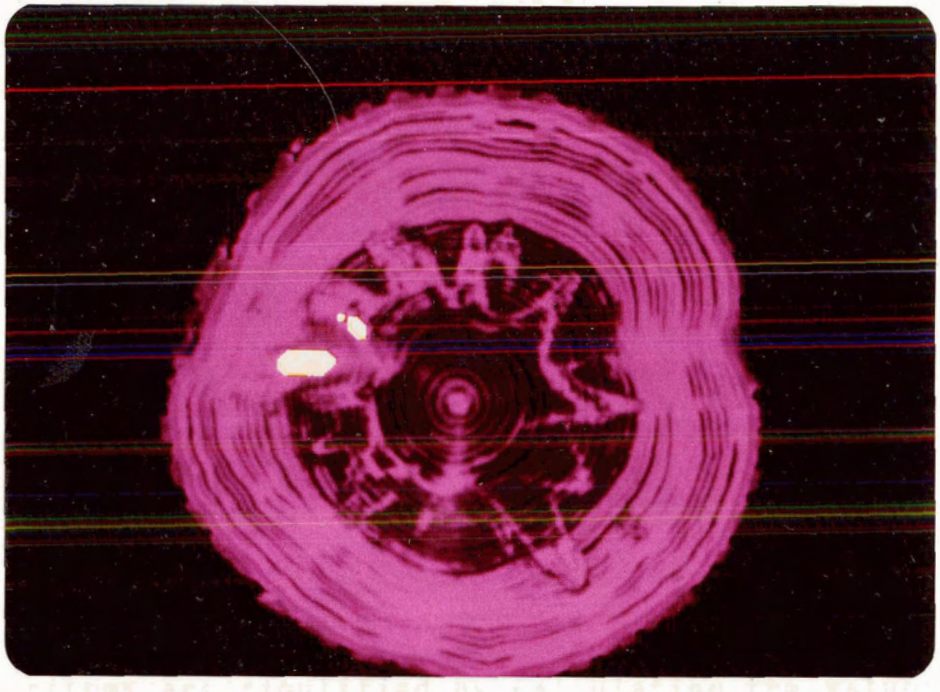


Figure 15.
Possible Knot Regions
after Region Growing
(log sample of green hemlock)
The original CT-scan is displayed in magenta. The high density areas, now with uniform shape, are displayed in white.

Figures 14 and 15 show the possible knot regions (as determined by histogram analysis) grown/pruned into convex regions.

4.2.2. Region Processing for Knot Measure

The previously grown regions are processed further to determine a measure of the possibility that the region corresponds to a knot. Because of the previous region growing, the size and shape of regions is easily specified. Each can be represented as a rectangle with a right isoceles triangle (of possibly zero size) "missing" from each corner. A processed region can be completely specified by the height and breadth of its bounding rectangle, the location in the image of the upper left corner of its bounding rectangle, and the size of each of the "missing" corners (given as the length of the equal side of the missing corner). Although the motivation for the following calculations comes from Baumgart [1974], the actual algorithms are adapted from Winston [1981]. The algorithms of Winston are specified for a single binary region of arbitrary shape. They are modified to process regions with standard convex shape. The algorithms are simplified by calculating the values required for the bounding rectangle, generally a simple task, and then subtracting the values for each of the "missing" corners. By simplifying these algorithms algebraically, we reduce the computations required to a function of eight variables. Therefore, this is an implementation which will calculate the required values in constant time, regardless of region size. Details of the calculations performed here are provided in Appendix A.

Using the above algorithms, the following values are calculated: the area of the region, the center of mass of the region, the minimum and maximum inertia about the center of mass of the region, and the angle of the principal axis of inertia. The elongation factor of the region is calculated as the difference of the minimum and maximum inertia divided by their sum:

$$\text{elong. fact.} = \frac{\text{max. inertia} - \text{min. inertia}}{\text{max. inertia} + \text{min. inertia}} \qquad (2)$$

The elongation factor of a circle is zero. The maximum value possible for an elongation factor is one.

Recall that a knot is roughly elliptical in shape, with its major axis pointing towards the center of growth. The knot measure is a measure of how nearly a particular region satisfies the constraint of pointing towards the center of growth. A simple knot measure is the angular difference between the direction of the principal axis of inertia and the line from the center of growth through the center of mass of the region. For nearly circular regions (nearly octagonal regions in our representation) the relative error in the direction of the principal axis of inertia may be large. Therefore we scale the angular difference by the elongation factor so that the more circular the shape of a particular region, less weight is given to the angular difference between the principal axis of inertia and the line from the center of

growth through the center of mass of the region.

Since knots tend to occupy a significant local area of a CT-scan of a log, region area is used to prune out regions which may not be knots. Regions which are under 10 pixels in area are not considered as knots.

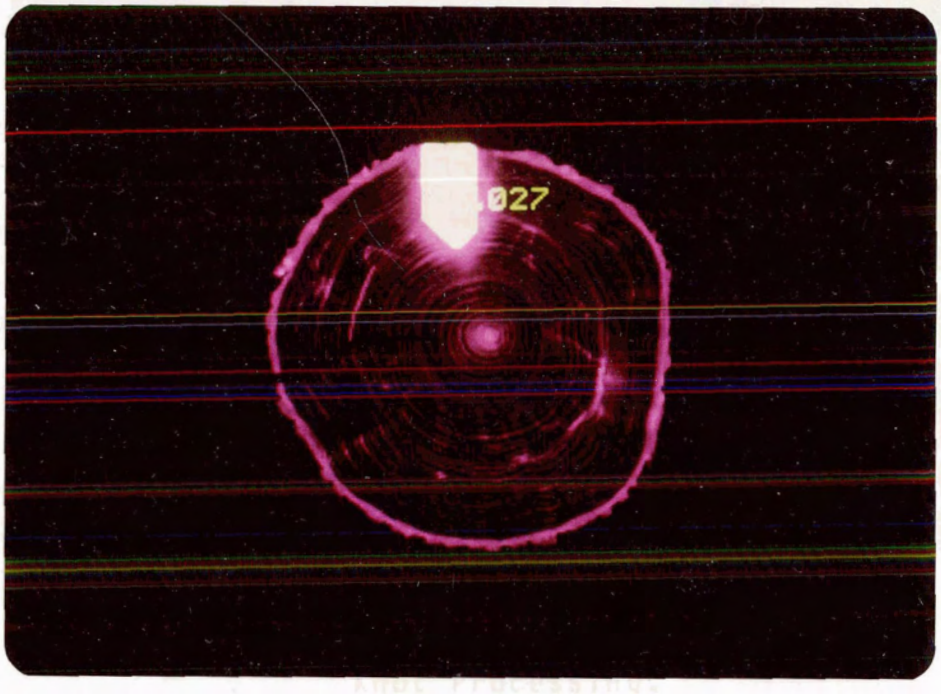


Figure 16.
Results of Possible
Knot Processing.
(log sample of aged hemlock)
The original CT-scan is displayed in magenta. The high density areas are displayed in white. The principal axis of inertia are displayed, together with the calculated knot measures. Here a knot measure of .027 indicates that this region is a knot.

Figure 16 shows the results of the knot processing. The region, its principal axis of inertia, and the calculated knot measure are shown. Experimentally, over the log samples of

various species analysed, a knot measure of less than approximately 0.3 indicates regions that are knots, while a knot measure of greater than 0.3 indicates high density regions that do not correspond to knots.

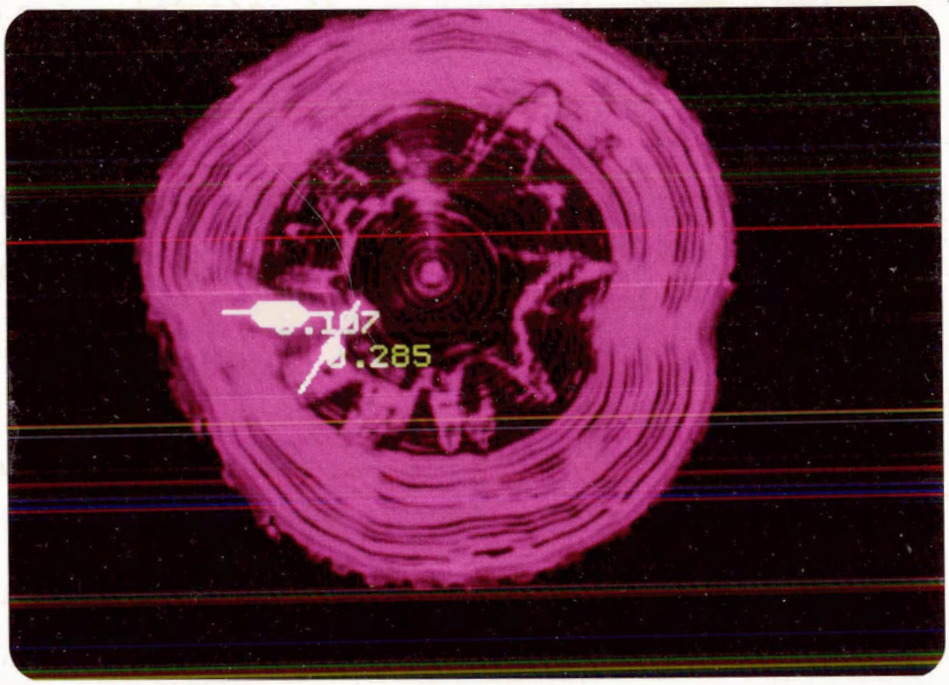


Figure 17.
Results of Possible
Knot Processing.
(log sample of green hemlock)
The original CT-scan is displayed in magenta. The high density areas are displayed in white. The principal axes of inertia is displayed, together with the calculated knot measure. Here knot measures of .107 and .285 indicate that these regions are knots.

Figure 17 also shows the results of region growing and possible knot region processing. Note that some of the sapwood which was initially found to be possible knots can be discounted because of the knot measure. That is, high density sapwood, when located as possible knots by the histogram analysis, is

typically a region parallel to the growth rings, and hence perpendicular to regions corresponding to knots. Also, many of the small possible-knot regions generated by the region grower have been eliminated because of their small area.

4.3. Possible Good Wood Analysis

The remainder of the CT-scan of a log, after the low and high density regions have been identified, is initially taken to be an area of good wood, with the possible exception of some rot. Good wood contains uniform concentric growth rings which show up distinctly in CT-scans. Rotting wood, on the other hand, causes a breakdown in the internal structure of a log, and thus destroys the growth rings. This difference, in combination with density information, is used to identify rotten wood.

The growth rings in a CT-scan of a log present a locally uniform texture. Since growth rings are destroyed by rot, a possible measure of good wood is the uniformity of the growth ring texture. To compute this texture measure, the growth rings in the CT-scan of a log are first found by applying an edge detector. The resulting edge elements are then interpreted to determine a directional uniformity measure. This structural measure of texture is similar to the work of Nevatia [1979], Nevatia [1981], and Tomita [1979].

4.3.1. Growth Ring Edge Detection

In order to process the regions of possible good wood, it was necessary to first determine the edge elements in the CT-scan of the log. The edge detector used here was first proposed by Marr and Hildreth (Marr [1979]). Edges are located at the zero-crossings of the Laplacian (sum of the unmixed partial second derivatives) of the smoothed original CT-scan image. Taking the Laplacian of the smoothed CT-scan image is equivalent (Castleman [1979]) to convolving the original CT-scan image with the Laplacian of the two-dimensional Gaussian smoothing function, and this method is employed here. After convolving the CT-scan with the Laplacian of the Gaussian function, the zero-crossings must be located. Zero-crossings are the boundaries between the positive and negative regions in the convolved image. Zero-crossings are found by convolving the result of the first convolution with a small mask which locates those pixels with a non-negative value (positive or zero) which have pixels with a negative value above, below, or beside them.

This edge detector is implemented on the IIS in two stages. In the first stage, the original image is convolved with the Laplacian of a two-dimensional Gaussian function. In the second stage, the zero-crossings between the positive and negative regions of this convolved image are determined.

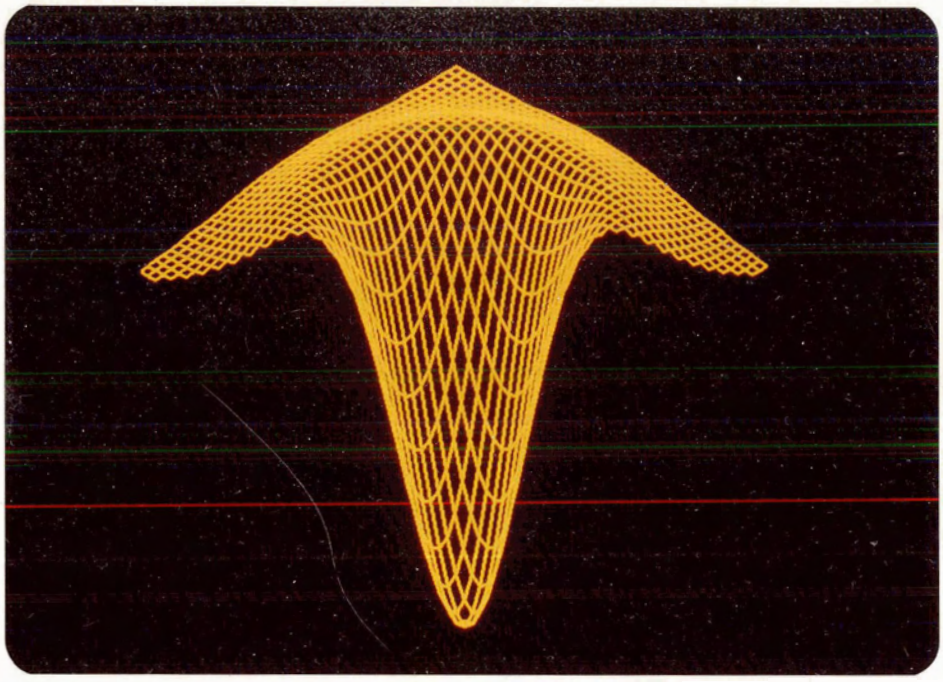


Figure 18.
Cross-Section of the Laplacian of a Two-Dimensional Gaussian Function Convolution with this mask is used to determine the second derivative of the of the original CT-scan.

The Laplacian of a two-dimensional Gaussian function has the shape of a inverted Mexican hat, as shown in Figure 18. It can be pre-computed, and is stored as a two-dimensional array of floating point values. The result of this convolution is a 16-bit signed-value image in channels 1 and 2. The next stage is to determine the zero-crossings, and hence the edge elements.

For edge detection, we are only interested in the sign of the result of the convolution with the Laplacian of the two-dimensional Gaussian function, since the zero-crossings are the boundaries between the positive and negative regions. The

magnitude of the convolved image will be used later. An edge element is defined as a pixel in a non-negative region of the convolved image which has, as a 4-connected neighbour, a pixel in a negative region of the convolved image. Edge elements can be found as follows:

- write a LUT to create a binary image of the convolved image, such that a zero indicates a non-negative region, and a one indicates a negative region.
- convolve this binary image with the mask:

0	1	0
1	0	1
0	1	0

  to create a possible edge element image.
- rewrite the original LUT so that non-negative regions have value 255, and negative regions have value 0, to create a non-negative region image.
- take the logical AND of the non-negative region image and the possible edge element image.
- the edge element image contains values from zero to four. Write a LUT to map the values from 1 to 4 to 1.

The third and fourth steps are necessary, since the mask

applied to the binary image will have non-zero values in negative regions, and these must be eliminated. The logical AND of the two images is accomplished using the same feedback mechanism used in image convolution, except that the alu function is changed. Thus, the logical AND of the two images is accomplished in one video frame time. The complete edge detection operator executes in roughly 15 seconds, depending on the size of the Gaussian mask. This speed is significant when one considers the 31,719,424 floating point multiplications and 262,144 floating point divisions necessary to convolve a 512 by 512 image with an 11 by 11 mask without the IIS. These floating point operations alone take over 9 minutes to execute on the VAX 11/750.

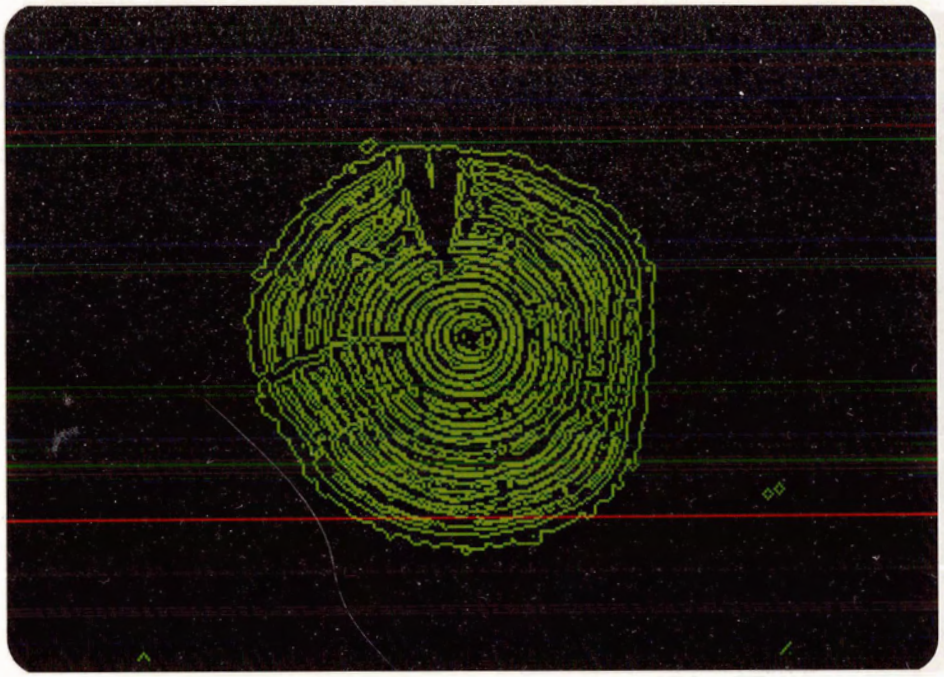


Figure 19.
Results of Edge Detection
(log sample of aged hemlock)
The edge elements generated by the edge detector are shown. Edge elements correspond to growth rings, and crack, knot, and log boundaries.

Figure 19 shows the result of applying the Marr and Hildreth edge detector to a CT-scan image of a log. The edge elements primarily correspond to growth rings, while a few correspond to the knot boundary.

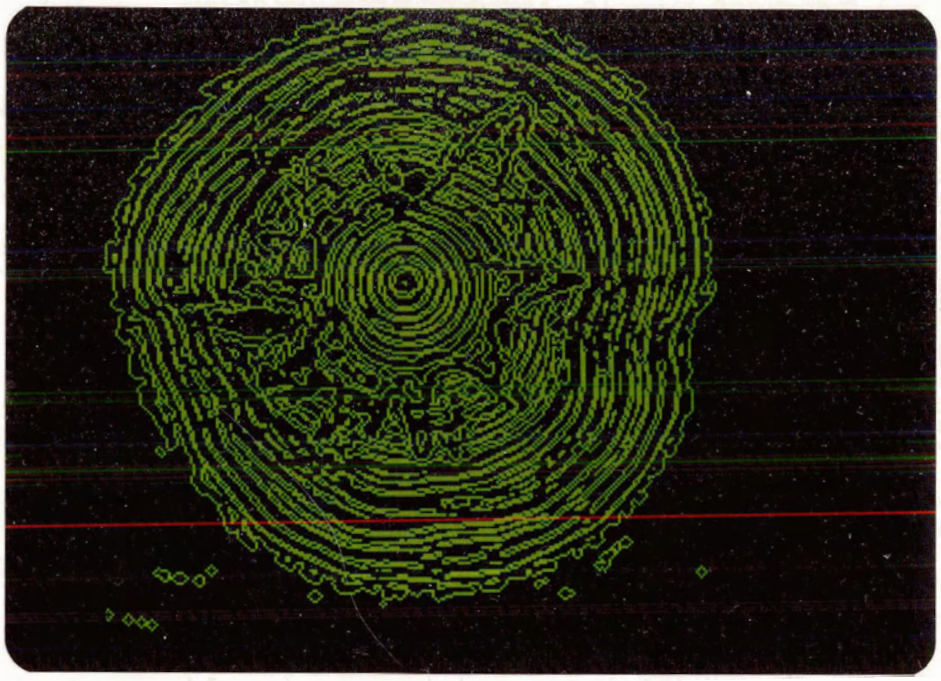


Figure 20.
Results of Edge Detection
(log sample of green hemlock)
The edge elements generated by the edge detector are shown. Edge elements correspond to growth rings, and crack, knot, and log boundaries.

In addition, in Figure 20, which shows the edge elements detected in a CT-scan image of green wood, some reflect the boundary between sapwood and heartwood, and the "sappy" areas (higher intensity) of the heartwood.

4.3.2. Growth Ring Edge Uniformity

We can use the edge elements we have found, together with the convolved image, to determine the 'uniformity' of growth rings. Such uniformity is an indicator of good wood. The heuristic which is implemented here may be described as follows. Good wood is evident in a CT-scan by the presence of

uniform, well-defined growth rings. Rot, and other abnormalities in the internal structure of a log will cause the growth rings to become discontinuous. They will also introduce edges (such as rot boundaries) into the CT image which are not parallel to the direction of the growth rings. For any given region, the maximum number of edge elements in any given direction divided by the total number of edge elements in that region is used as a measure of edge element uniformity. Since the growth rings are the only artifacts with uniform direction, this measure of edge element uniformity is also a measure of growth ring uniformity, and hence a measure of good wood. Within a small region, the edge elements to consider are those which are the non-terminating end of a linear group of edge elements, and non-zero in magnitude. An edge element is a non-terminating end of a linear group of edge elements if it has edge elements on any 2 opposite sides of it. Note that an edge element may be part of at most 4 linear groups of edge elements. The magnitude of an edge element is the difference, in the convolved image, between the 2 pixel values adjacent to the edge element perpendicular to the direction of the linear group of edge elements.

Uniformity calculation is implemented in the IIS as follows. Since the result of convolving the original CT-scan with the Laplacian of the two-dimensional Gaussian function is a 16-bit image, it is necessary to extract only the 8 most

significant bits. This is accomplished by feeding back the 16-bit two-channel convolved image onto itself, but on each feedback operation, shifting all the values right by 1 bit. After some number of operations (at most 7) the 8 most significant bits of the original convolved image are in the lowest byte of the data, and this can be stored in one channel.

To determine if an edge element is a non-terminating member of a linear group of edge elements, first treat the edge element image as a binary image, with a 1 corresponding to the presence of an edge element. Then apply the 8-connected neighbour encoding operator as in the convex region grower, storing the locations of the pixel neighbours in 8 bits. An edge element is a non-terminating member of a linear group of edge elements in a particular direction if bits 0 and 4, 1 and 5, 2 and 6, or 3 and 7 are on. Write a LUT for this neighbour encoding image, storing the directions for each edge element in bits 0, 1, 2, and 3.

To determine if there is a non-zero magnitude for each edge element, in each of the above 4 directions, convolve the previously saved 8-bit image with the following mask:

0	1	0
0	0	0
0	-1	0

rotated by 0, 45, 90, and 135 degrees. After each convolution,

take the absolute value of the result and form a binary image, with a one corresponding to a non-zero result, and a zero otherwise. These binary images are stored in bits 4, 5, 6, and 7, respectively, and ORed into the 4 bit direction image.

Thus, we have an 8-bit magnitude/direction image. The first 4 bits have a bit set if the magnitude of the edge element is non-zero in that direction, and the second 4 bits have a bit set if there is an edge in that corresponding direction. For each of the four directions (as above), set up a LUT to create a binary image, with a one if there is both a non-zero magnitude bit and non-zero direction bit for that direction, otherwise zero. Convolve this binary image with a mask of all ones, which counts up the number of occurrences of that magnitude/direction in the region. Compare this result with the result of the previous count (initially zero). Store the maximum of the two values. Next, determine the total number of magnitude/direction pairs in the region by writing a LUT which has values 0 through 4, depending on the number of magnitude/direction pairs in any edge element, and then convolve it with the same mask of all ones.

The remaining step in determining uniformity is to divide the maximum number of magnitude/direction pairs found by the total number in the region. This can be accomplished by writing a logarithmic LUT for the maximum image channel, a negative logarithmic LUT for the total image channel, and an

exponential OFM to display them through. Since displayed channels are added, the displayed result is the result of dividing the maximum number of magnitude/direction pairs in any one direction divided by the total number of magnitude/direction pairs.

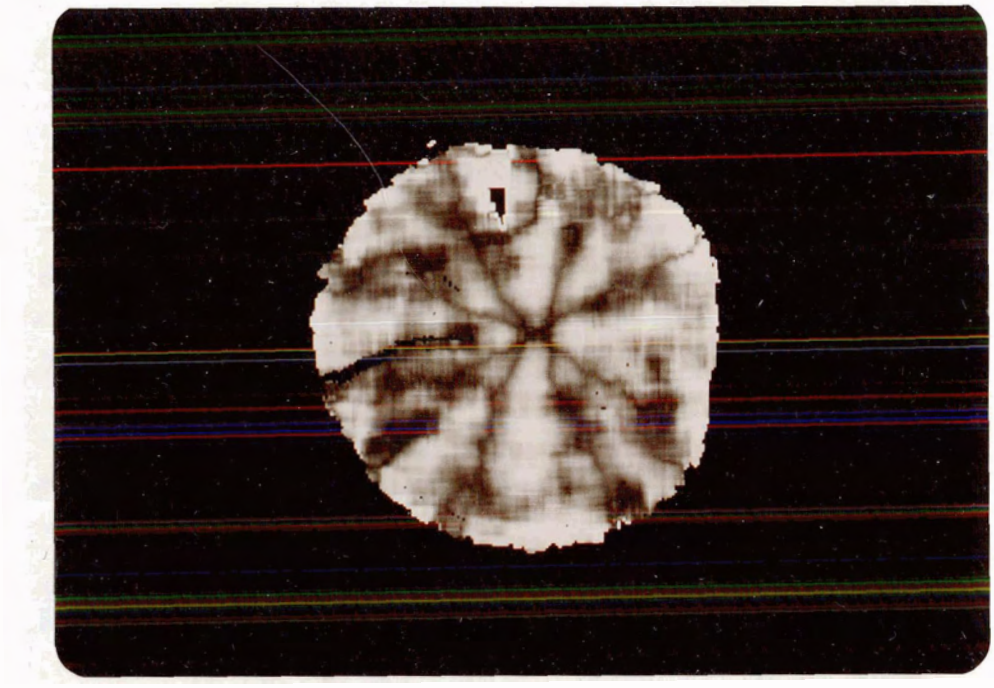


Figure 21.
Growth Ring Uniformity
(log sample of aged hemlock)
The result of the uniformity measure is shown as intensity. The higher the intensity, the higher the measured uniformity. The areas of low uniformity correspond to the knot and the crack on the left side. The radial lines through the center of growth are quantization errors and can be factored out later.

Figure 21 shows the results of this uniformity measure. Areas of higher intensity correspond to areas of good growth ring uniformity. The darker areas passing axially through the center of growth are merely an artifact of the quantization of

the directions into four possible values. This log has no rot, so the uniformity measure is relatively consistent across the image. Discontinuities may be noted, however, in areas adjacent to knots, but these knot regions are already accounted for.

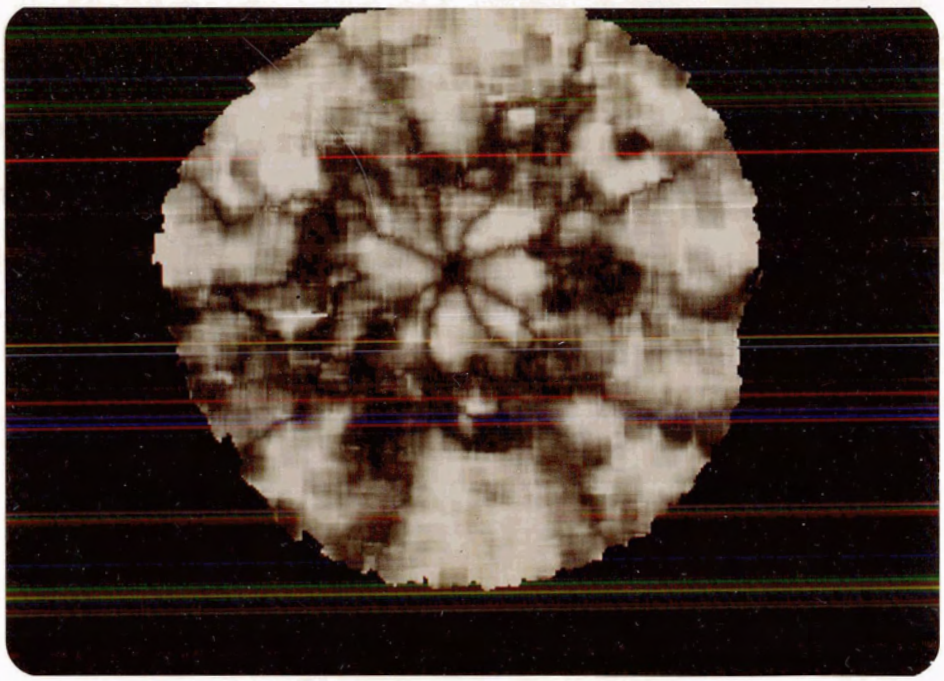


Figure 22.
Growth Ring Uniformity
(log sample of green hemlock)
The result of the uniformity measure is shown as intensity. The higher the intensity, the higher the measured uniformity. The radial lines through the center of growth are quantization errors and can be factored out later.

Figure 22 shows the result of applying the uniformity measure to green wood. Although there were many edge elements generated by the edge detector not corresponding to growth rings, the uniformity measure is nonetheless relatively consistent. This CT-scan also does not contain any rot.

Results of further CT-scan image interpretation, described in a later section, will illustrate the uniformity measure applied to CT-scans of log samples containing rot.

4.3.3. Determining Rotten Wood

Rotten wood has both low density and low uniformity. Good wood may have low density, but the growth ring structure generates a high uniformity measure. Conversely, good wood may have high density, but low uniformity (typically caused by closely packed growth rings not being detected by the CT-scanning process nor by the edge detector).

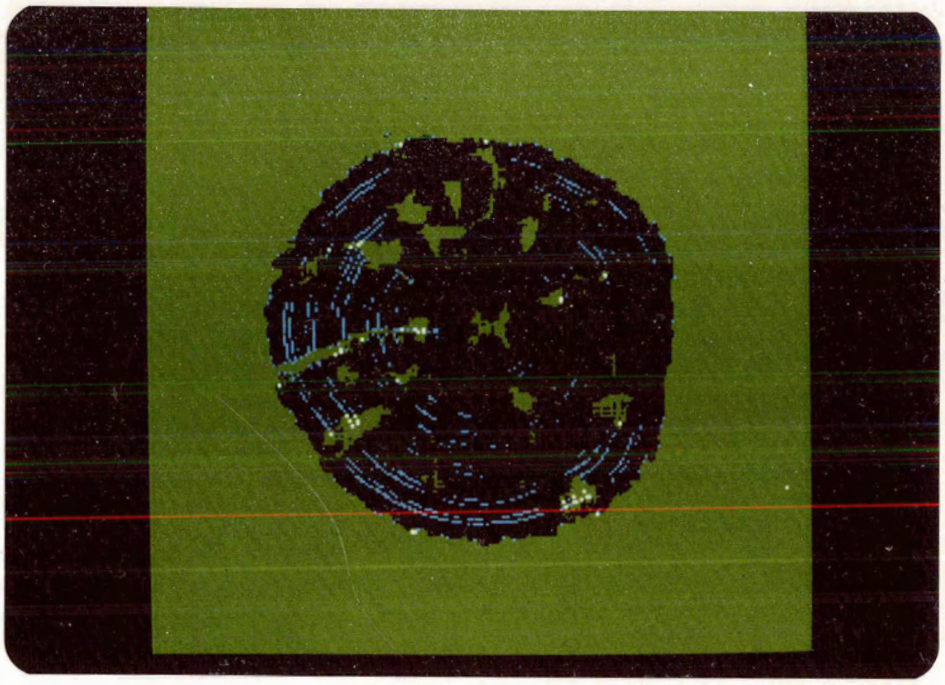


Figure 23.
Rotten Wood Indentification
(log sample of aged hemlock)
Areas of low uniformity are shown in green. Areas of low density are shown in cyan. Rot, the combination of low density and low uniformity, is shown in bright green.

Figure 23 shows the combination of low density and low uniformity. The low density threshold values were determined from the histogram analysis. The low uniformity threshold was obtained experimentally from the results of the previous section. Low density areas are shown in cyan. Low uniformity areas are shown in green. Rotten areas, the combination of these two, are shown in bright green. As previously noted, this log sample contains no rot, even though it has both low density and low uniformity regions.

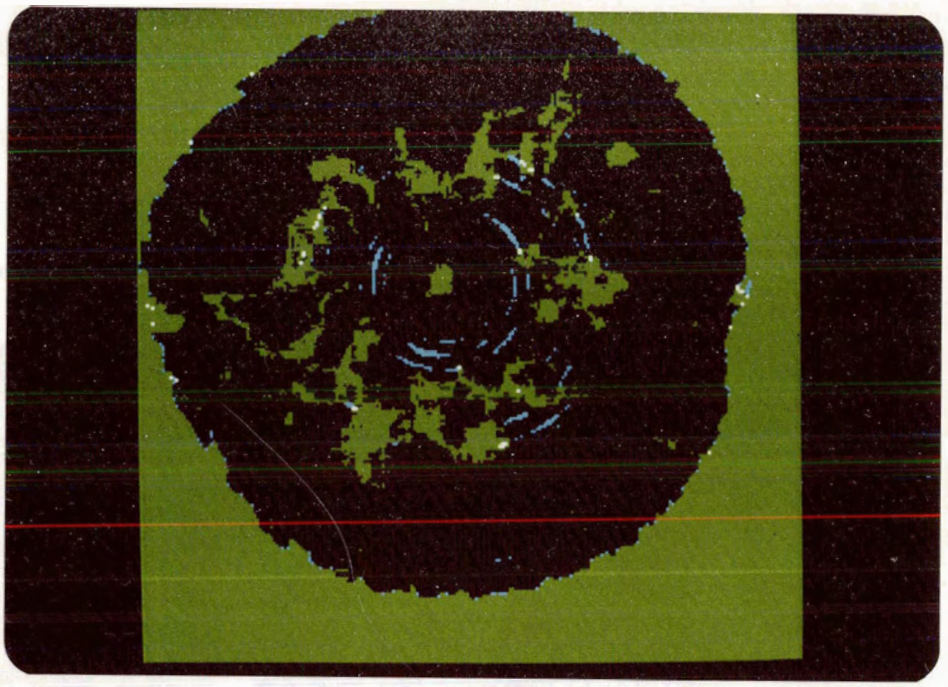


Figure 24.
Rotten Wood Identification
(log sample of green hemlock)
Areas of low uniformity are shown in green. Areas of low density are shown in cyan. Rot, the combination of low density and low uniformity, is shown in bright green.

Figure 24 shows the combination of low density and low uniformity for a sample of green wood. As in the previous sample, this log has no rotten wood.

As a final processing step, the rotten wood regions are grown using the same region growing operator used for knots. The results of this process are shown on a log sample containing rot in the next section.

5. Results

This section illustrates the interpretation results for log samples of various common species, with and without knots and rot.

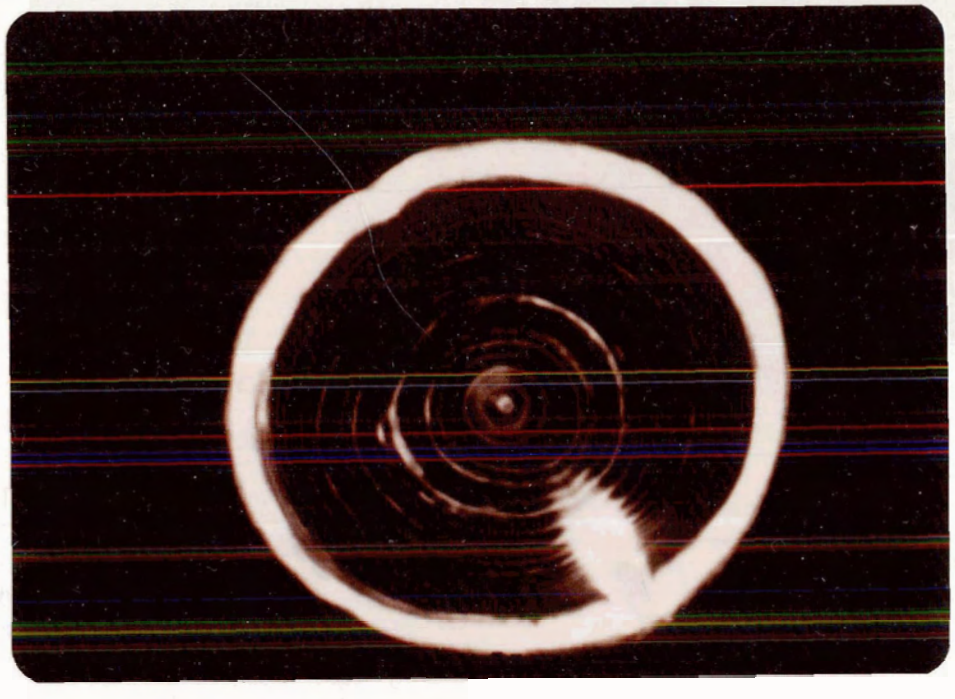


Figure 25.
CT-Scan of a Log
(log sample of green cedar with knots)
The CT-scan is displayed as a black and white image. Higher density areas, such as knots, are displayed as brighter regions.

Figure 25 shows the CT-scan of a green cedar log which contains no rot. This log has a narrow, but high density, band of sapwood surrounding the heartwood. There is a single knot in the lower right quadrant. Cedar is one of the least dense woods, but the interpretation algorithms successfully interpret the CT-scan.

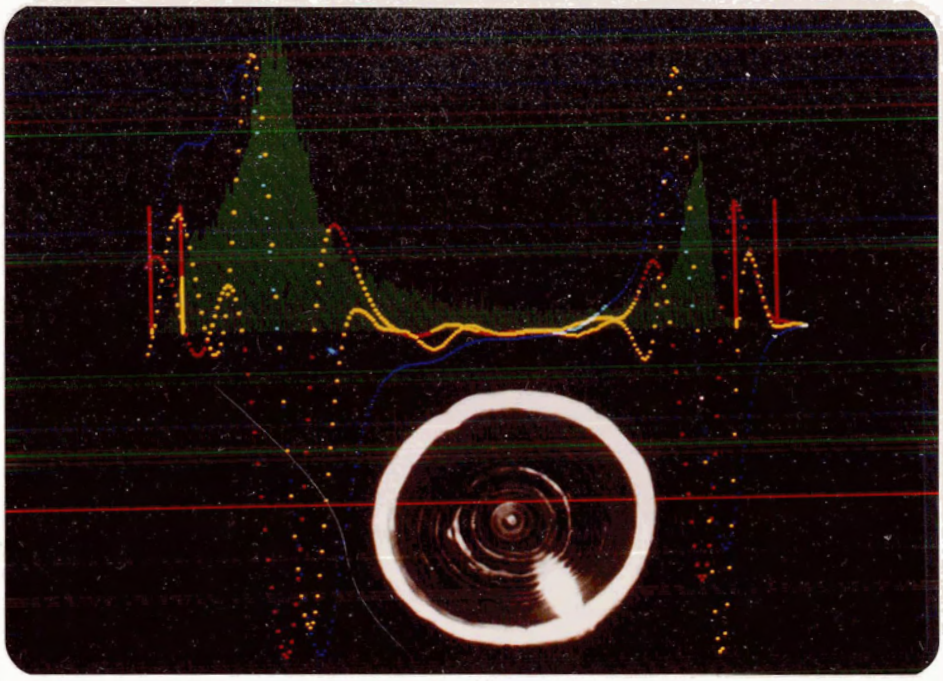


Figure 26.
Results of Histogram Analysis
(log sample of green cedar with knots)
The original histogram is shown in green. The first, second, and third derivatives of the histogram are shown in blue, red, and yellow, respectively. The four boundary points determined by histogram analysis are shown in red.

Figure 26 shows the histogram analysis of the CT-scan. Even though the sapwood has high density, the analysis detects the knot boundary. As in previous examples of histogram analysis. there is no significant knot peak in the histogram, so the rightmost zero-crossing is taken as the boundary between good wood and possible knots.

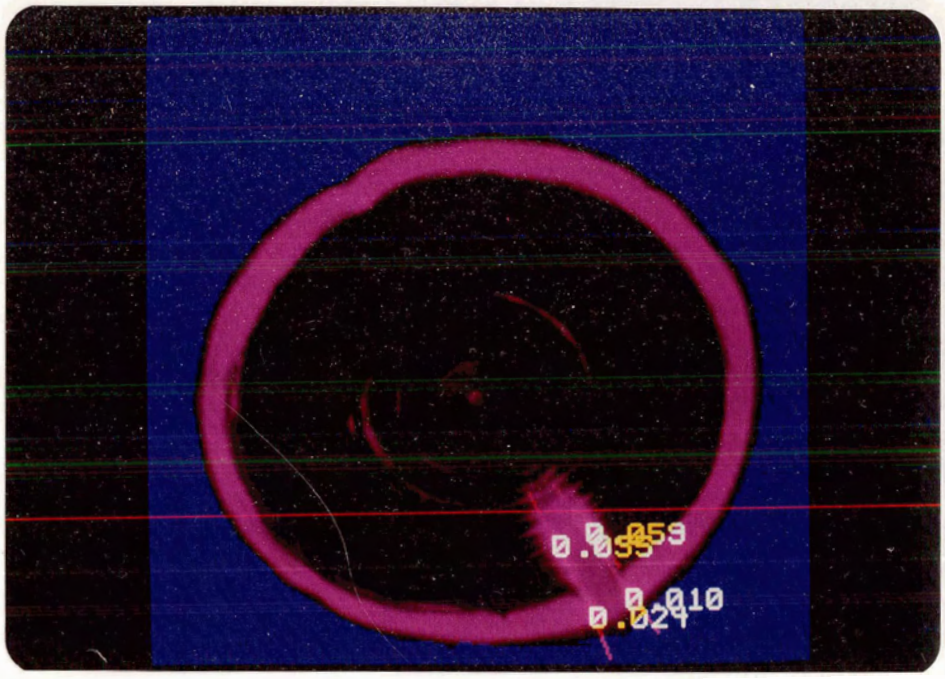


Figure 27.
Final Interpretation
(log sample of green cedar with knots)
Air, and other low density regions are displayed in blue. Good wood is displayed in magenta. Rot is displayed in green. The knots and the calculated knot measures are also displayed.

Figure 27 shows the final results of the interpretation of the CT-scan of the log. Regions of low density, corresponding to air and internal holes, are displayed in blue. Knots, with their knot measure, are displayed in red. Of the remainder of the CT-scan, rot is indicated by green areas, and good wood appears, in magenta, as it did in the initial CT-scan. Also shown are the results of determining the possible knot measure for the convex regions in the lower right portion of the CT-scan. As shown by the knot measures of less than 0.3, this regions are correctly identified as knots.

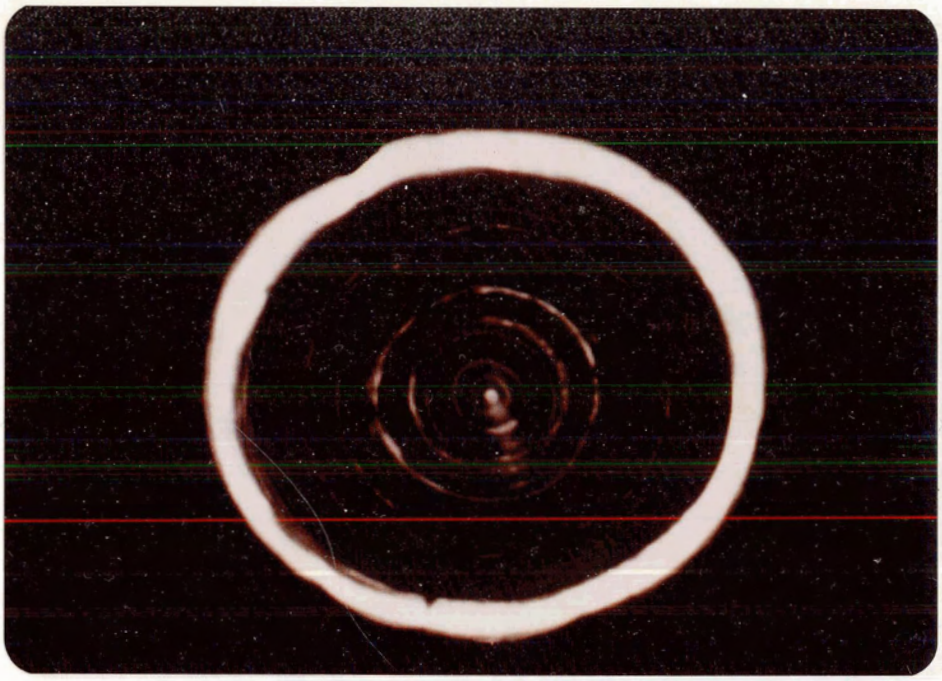


Figure 28.
CT-Scan of a Log
(log sample of green cedar without knots)
The CT-scan is displayed as a black and white image. Higher density areas, such as knots, are displayed as brighter regions.

Figure 28 shows the CT-scan of another sample of green cedar, taken from the same tree as the previous CT-scan, except this log sample does not contain any knots. The interpretation process, in particular the histogram analysis process, can accommodate the absence of knots.

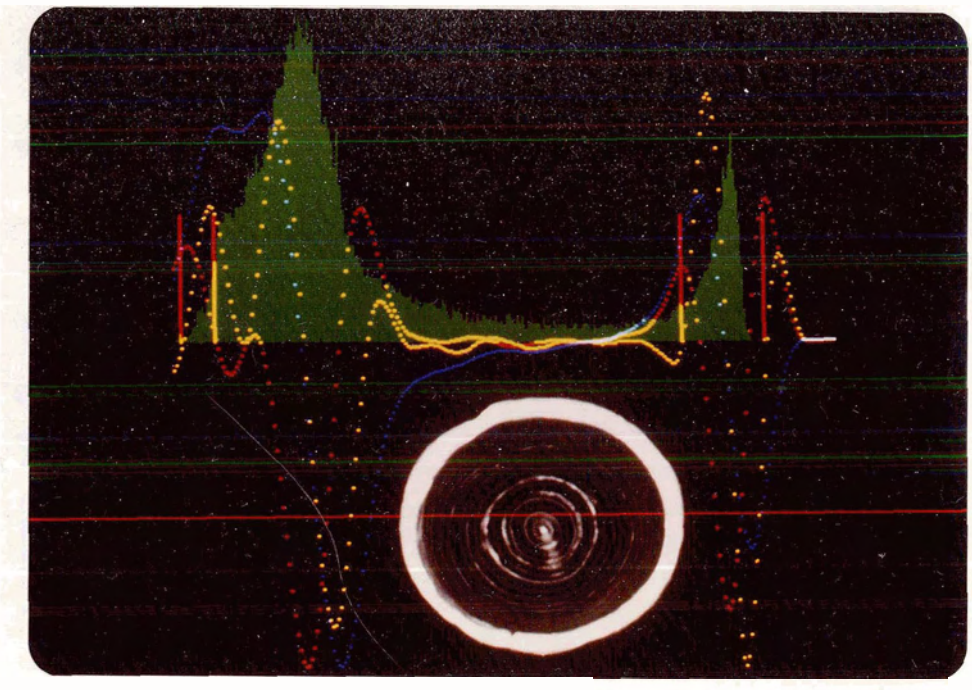


Figure 29.
Results of Histogram Analysis
(log sample of green cedar without knots)
The original histogram is shown in green. The first, second, and third derivatives of the histogram are shown in blue, red, and yellow, respectively. The four boundary points determined by histogram analysis are shown in red.

Figure 29 shows the result of histogram analysis of the CT-scan. Note that even though there are no knots, the possible knot threshold is nearly the same value as in the previous cedar log sample with knots.

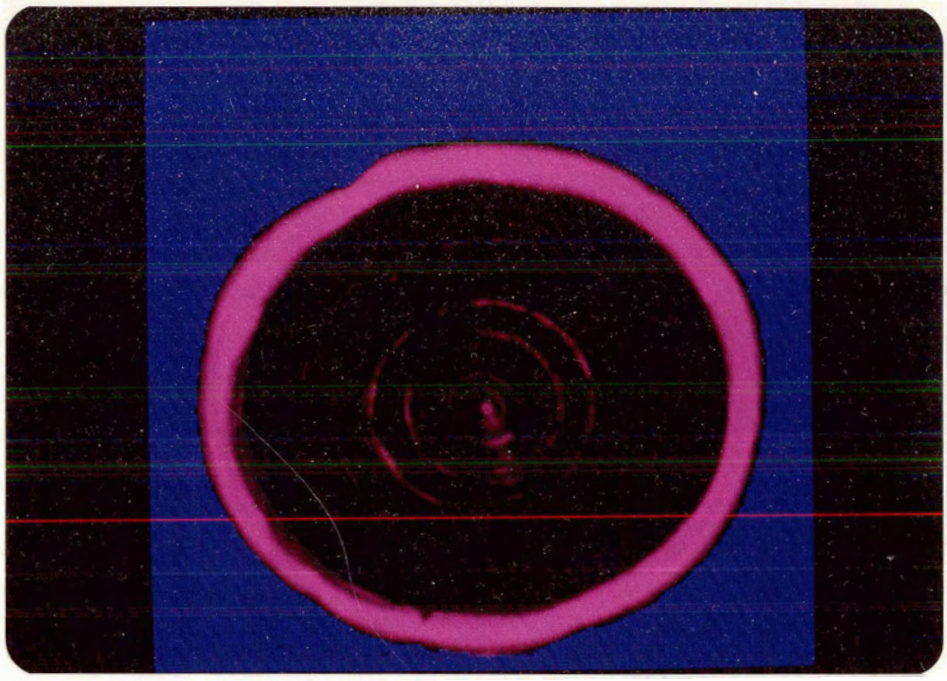


Figure 30.
Final Interpretation
(log sample of green cedar without knots)
Air, and other low density regions are displayed in blue. Good wood is displayed in magenta.

Figure 30 shows the results of the complete interpretation process. The results displayed here are very similar to the results of the previous CT-scan of a green cedar log, with the exception of the knot.

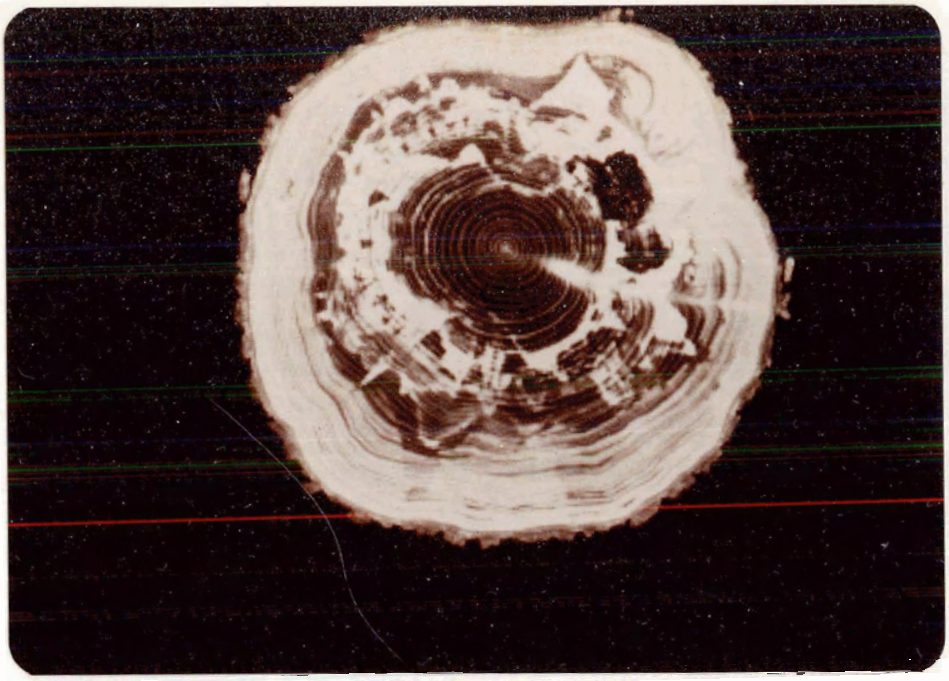


Figure 31.
CT-Scan of a Log
(log sample of green hemlock with rot)
The CT-scan is displayed as a black and white image. Higher density areas, such as knots, are displayed as brighter regions.

Figure 31 is a CT-scan of a green hemlock log containing a rotten region in the upper right quadrant. In addition, this log also contains a knot in the lower right quadrant.

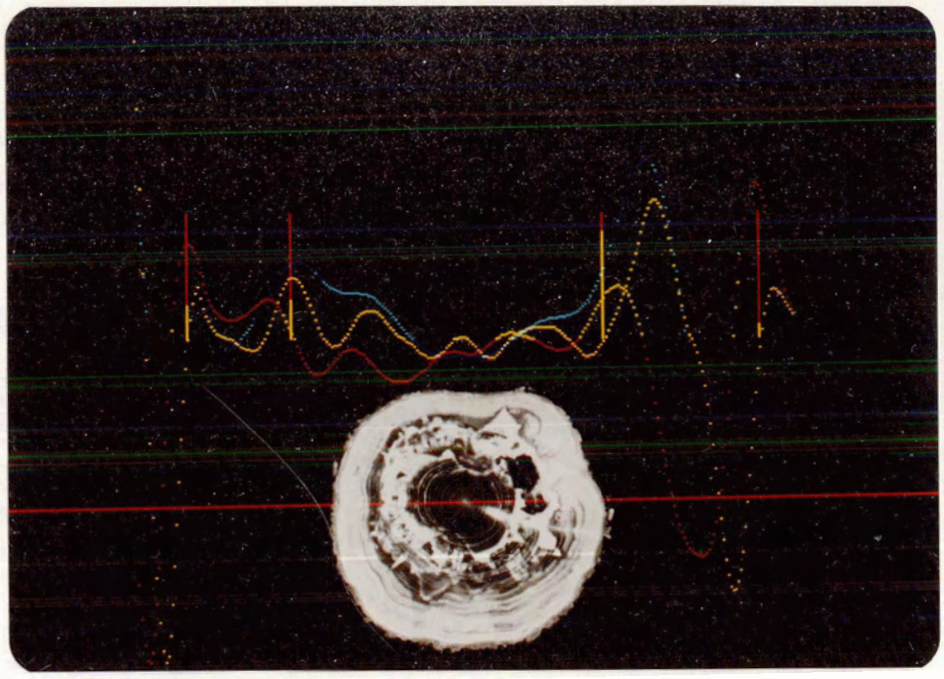


Figure 32.
Results of Histogram Analysis
(log sample of green hemlock with rot)
The original histogram is shown in green. The first, second, and third derivatives of the histogram are shown in blue, red, and yellow, respectively. The four boundary points determined by histogram analysis are shown in red.

Figure 32 shows the histogram analysis. The rotten region has density values in the center low enough to indicate a hole, but also density values near the edge of the regions high enough to be included as possible good wood. However, these values are within the range of possibly rotten good wood determined by the histogram analysis.

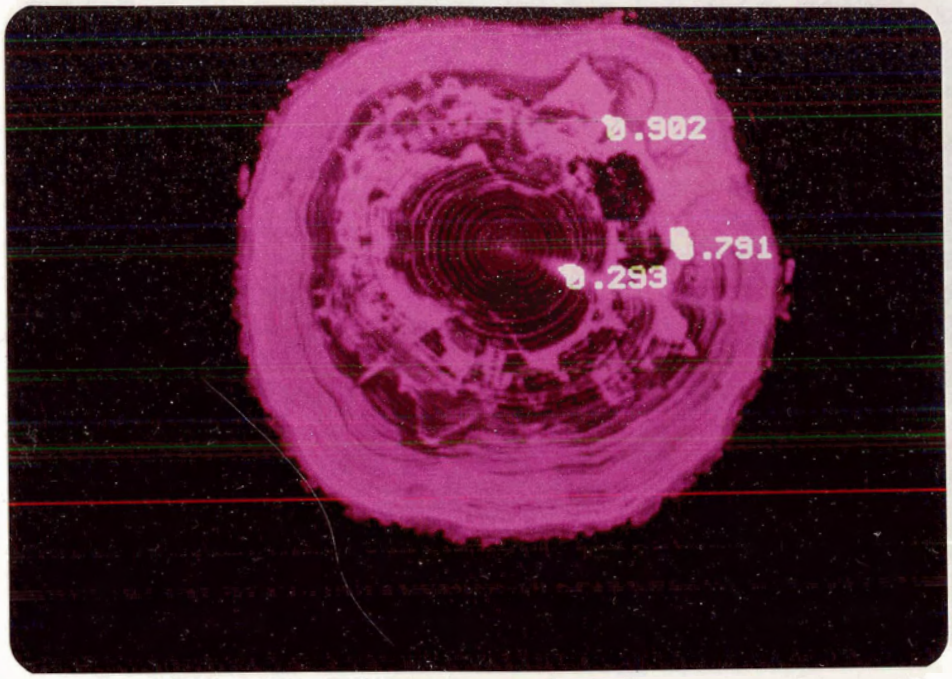


Figure 33.
Results of Possible Knot Analysis
(log sample of green hemlock with rot)
The original CT-scan is displayed in magenta. The high density areas are displayed in white. The principal axes of inertia are displayed, together with the calculated knot measure. Here knot measures of .902 and .791 indicate that these regions are not knots, while the other knot meeasure of .293 indicates a knot.

Figure 33 shows the knot analysis. Many regions, generated by thresholding using the value for possible knots from histogram analysis, have been pruned out during the calculation of the knot measure due to their small area.

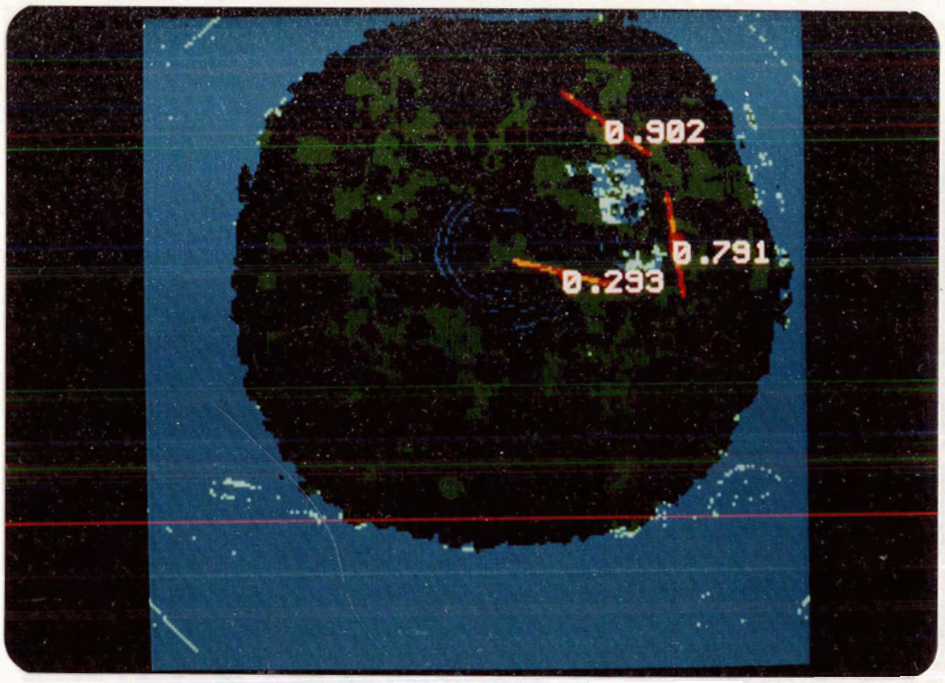


Figure 34.
Results of Low Density and Low Uniformity Measure
(log sample of green hemlock with rot)
Areas of low uniformity are shown in green. Areas of low density are shown in cyan. Rot, the combination of low density and low uniformity, is shown in bright green.

Figure 34 shows the combination of low uniformity areas, and low density possible good wood areas. There are low density good wood areas detected between the growth rings, particularily near the center of growth, but the uniformity is high (due to the good growth ring structure). There are low uniformity areas detected in the region of sapwood, primarily due to the lack of discernible edge elements, but the density is too high to be rot. The region containing the rot has both low density and low uniformity, as can be seen in Figure 34. The central area of this region of rot has a density value similar to that of air and is indicated as a hole. This hole

in the rot appears in the final result.

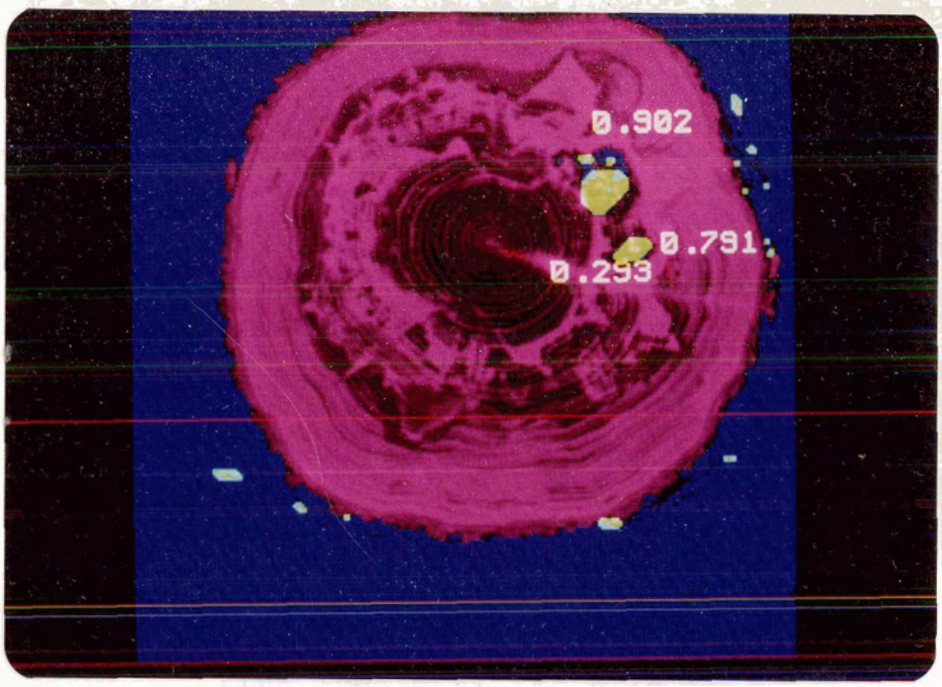


Figure 35.
Final Interpretation
(log sample of green hemlock with rot)
Air, and other low density regions are displayed in blue. Good wood is displayed in magenta. Rot is displayed in green. The knots and the calculated knot measures are also displayed.

Figure 35 shows the final result of interpreting the log sample of green hemlock containing some rot. As can be seen, the rotten region is indicated as a hole, surrounded by a region of rotten wood. This is consistent with the breakdown of the internal structure of a log, where rotten wood eventually decays until a hole is created. The knot measure can be used to prune out all possible knot regions, with the exception of the true knot.

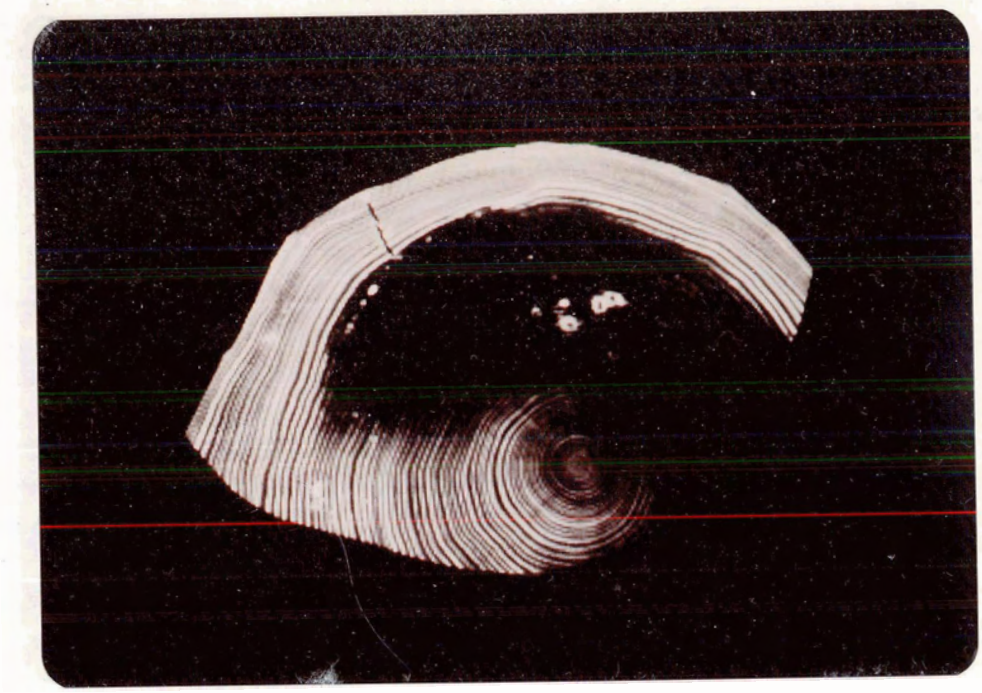


Figure 36.
CT-Scan of a Log
(log sample of fir with rot)
The CT-scan is displayed as a black and white image. Higher density areas, such as knots, are displayed as brighter regions.

Figure 36 is a CT-scan of a fir log containing extensive rotten regions. This log sample was too large to be CT-scanned and was cut longitudinaly prior to scanning.

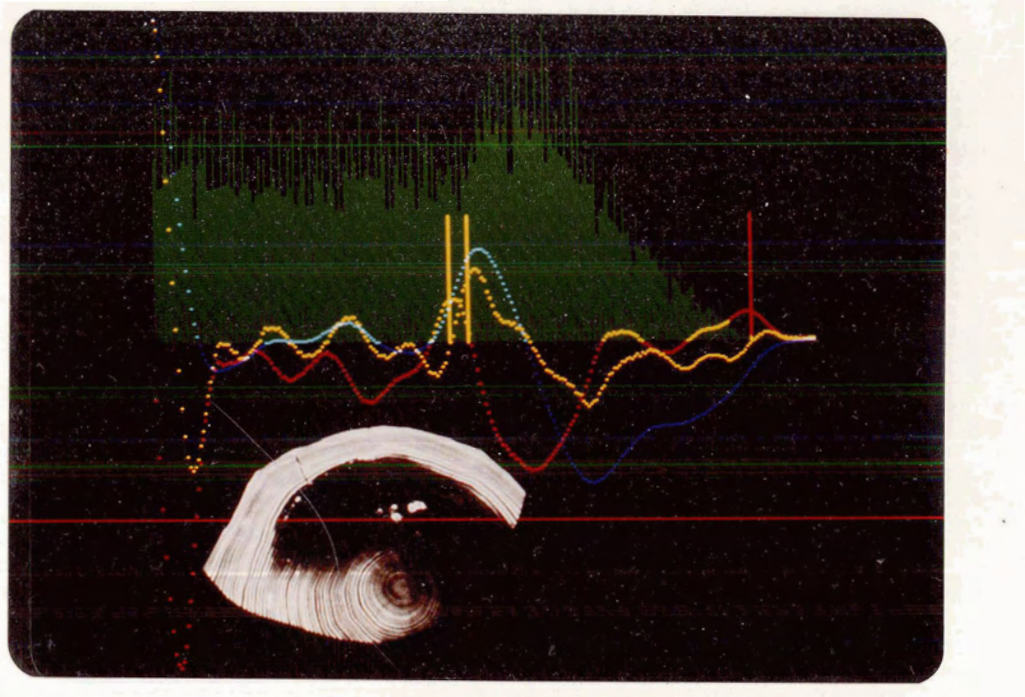


Figure 37.
Results of Histogram Analysis
(log sample of fir with rot)
The original histogram is shown in green. The first, second, and third derivatives of the histogram are shown in blue, red, and yellow, respectively. The four boundary points determined by histogram analysis are shown in red.

Figure 37 shows the histogram analysis. The extensive rotten region is determined primarily by its low density.

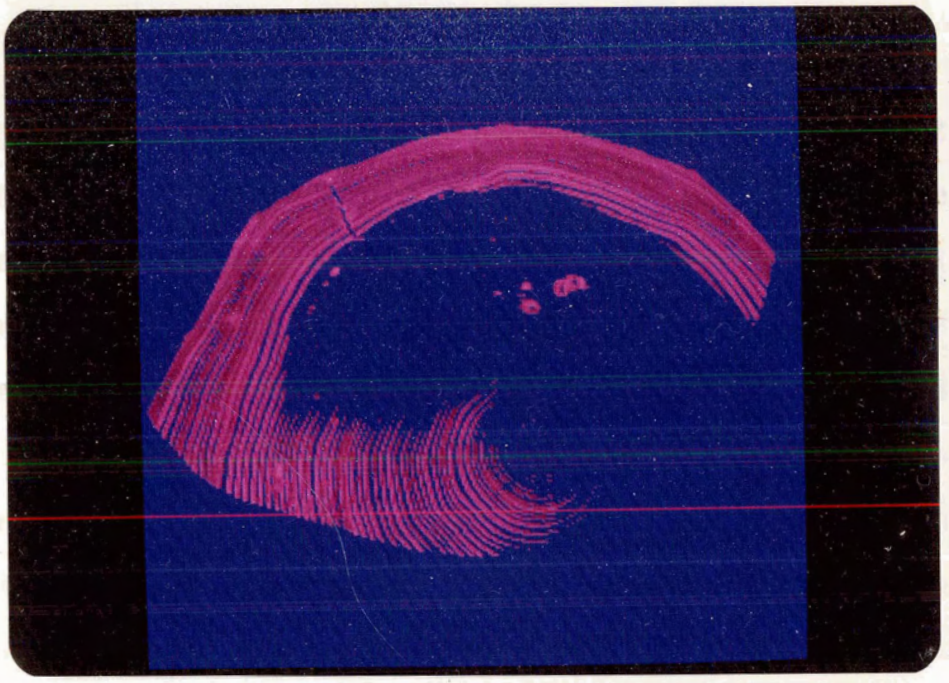


Figure 38.
Final Interpretation
(log sample of fir with rot)
Air, and other low density regions are displayed in blue. Good wood is displayed in magenta. Rot is displayed in green The knots and the calculated knot measure are also displayed.

Figure 38 shows the final result of interpreting the rotten fir log sample. This log contains such extensive rot that it would not typically be present in the sawmilling environment. However, CT-scans of heavily damaged logs can be interpreted by these algorithms. Small regions of marginal good wood identified as rot were pruned by the convex region grower.

6. Conclusions

Algorithms are developed in this thesis which interpret CT-scan images of log samples. The algorithms combine a number of image-processing techniques, such as histogram analysis, and texture analysis, directed by knowledge of the internal structure of logs. The implementation of these algorithms used, in part, a digital image processor. Results were presented showing the application of the interpretation algorithms on a variety of log samples. These log samples were of different species and age. This interpretation process can occur quite efficiently through the use of special purpose digital image processing hardware. This thesis is a successful first step in the total computerization of determining optimal sawlog utilization. However, further work needs to be done on: developing an industrial CT-scanner, capable of continuous operation in an industrial environment; improving the execution speed of the algorithms outlined in this thesis; and creating a three-dimensional model of the internal structure of a log from a number of adjacent CT-scans, and manipulating this representation to create an optimal cutting strategy.

7. Appendix A

Our purpose in this appendix is to show the derivation of the functions which compute the required values for the convex knot regions. First, let us look at the equations of Winston [1981] which we will simplify. Assume that our image contains only one region, represented as a binary image. Pixel values, denoted P_{ij} (pixel value at coordinates i and j), are either zero if outside the region, or one if inside. The area, A, of the region is simply:

$$A = \sum_i \sum_j P_{ij} \tag{3}$$

The center of mass of a region, at coordinates I_o and J_o, is calculated as follows:

$$I_o = \frac{1}{A} \sum_i \sum_j i*P_{ij} \tag{4}$$

and

$$J_o = \frac{1}{A} \sum_i \sum_j j*P_{ij} \tag{5}$$

We can define a new coordinate system centered about the center of mass:

$$i' = i - I_o \tag{6}$$

and

$$j' = j - J_o \tag{7}$$

The perpendicular distance from any point in the region to an axis passing through the center of mass of the region is:

$$perp\_dist = i'*\sin(\theta) - j'*\cos(\theta) \tag{8}$$

where θ is the angle of this axis from the horizontal. The inertia, I, of this region is simply the sum of the square of this perpendicular distance for all pixels in the region:

$$I(\theta) = \sum_i \sum_j P_{ij}*(i'*\sin(\theta) - j'*\cos(\theta))^2 \tag{9}$$

Expanding the squared term and simplifying yields:

$$I(\theta) = U*\sin^2(\theta) - V*\sin(\theta)*\cos(\theta) + W*\cos^2(\theta) \tag{10}$$

where:

$$U = \sum_i \sum_j i^2*P_{ij} - A*I_o^2 \tag{11}$$

$$V = \sum_i \sum_j i*j*P_{ij} - 2*A*I_o*J_o \tag{12}$$

$$W = \sum_i \sum_j j^2*P_{ij} - A*J_o^2 \tag{13}$$

Differentiating the above and solving for zero determines that the inertia is maximal when:

$$\tan(2*\theta) = \frac{V}{(U-W)} \tag{14}$$

or

$$\theta = \frac{1}{2} * \tan^{-1}\left(\frac{V}{(U-W)}\right) \tag{15}$$

As well, using U, V, and W, we can calculate the maximum inertia, I_{max}, and the minimum inertia, I_{min}:

$$I_{max} = (U+W) + \sqrt{V^2+(U-W)^2} \tag{16}$$

$$I_{min} = (U+W) - \sqrt{V^2+(U-W)^2} \tag{17}$$

Our method here is to calculate U, V, and W, and then determine the angle of the principal axis of inertia, as well as the values of the maximum and minimum inertia.

Our simplification of these algorithms can be described as follows. For every region in the image, there is a bounding rectangle. This is a rectangle formed by extending the top, bottom, and vertical sides of the region. Since the remaining sides of the regions are diagonal (oriented at 45, 135, 225 and 315 degrees), a region can be thought of as its bounding rectangle, less a right isoceles triangle at each corner. The measure of this triangle will be the length of one of the equal sides. For example, here is a typical region, with its bounding rectangle:

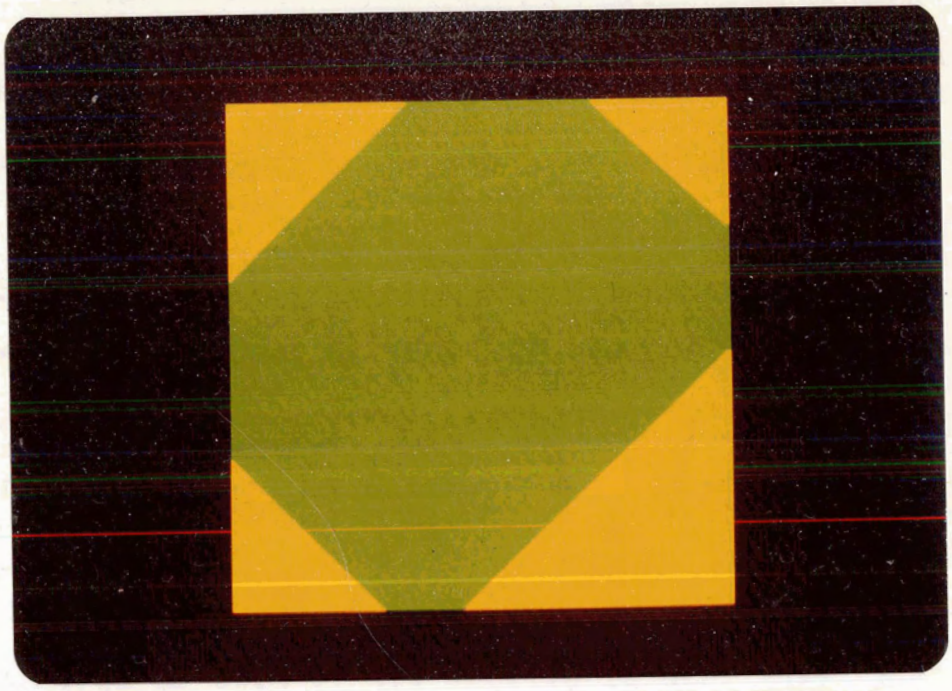


Figure 39.
A Typical Region with
Its Bounding Rectangle

From this figure, we can define the following quantities:

x, y - the coordinates of the upper left corner of the bounding rectangle of a region.

m, n - the height and width, respectively, of the bounding rectangle.

k_0, k_1, k_2, k_3 - the size of the missing corners. These values are the length of the equal side of the right isoceles triangle which forms the missing corner. The corners are numbered in a clockwise manner, with corner 0 at the upper left.

These 8 values completely specify the location and boundaries of a region in the image are used to calculate the values U, V, and W. Each of the values needed is first calculated for the bounding rectangle, and then the value for each of the corners is subtracted. As an example of this calculation simplification, consider calculating the area of a typical region. It is the area of the bounding rectangle:

$$A_r = m * n \tag{18}$$

less the area of each of the missing corners (say corner 0):

$$A_0 = \frac{k_0 * (k_0+1)}{2} \tag{19}$$

The equation is the same for calculating A_1, A_2, and A_3 from k_1, k_2, and k_3. Thus, the area of a region, A, is:

$$A = A_r - \sum_{i=0}^{3} A_i \tag{20}$$

Using the above method, the center of mass of a region is calculated as follows. First calculate I_o:

$$I_o = \frac{1}{A} \sum_i \sum_j i * P_{ij} \tag{21}$$

By translating the bounding rectangle to the origin, the double sum becomes (for the bounding rectangle):

$$C_r = (m+1) * m * \frac{n}{2} \tag{22}$$

for the upper corners (k_0 and k_1):

$$C_0 = \frac{(k_0-1) * k_0 * (k_0+1)}{6} \tag{23}$$

$$C_1 = \frac{(k_1-1) * k_1 * (k_1+1)}{6} \tag{24}$$

and for the lower corners (k_2 and k_3):

$$C_2 = \frac{(m-1)*A_2}{2} - \frac{(k_2-1)*k_2*(k_2+1)}{6} \tag{25}$$

$$C_3 = \frac{(m-1)*A_3}{2} - \frac{(k_3-1)*k_3*(k_3+1)}{6} \tag{26}$$

Thus:

$$I_o = x + \frac{C_r - (\sum_{i=0}^{3} C_i)}{A} \tag{27}$$

The analysis is the same for calculating J_o, except the distinction is made between the left and the right corners of the bounding rectangle, rather than the top and bottom corners.

Using the above method, we now calculate the values U, V, and W, and hence the axis of inertia. Recall the equation for U:

$$U = \sum_i \sum_j i^2 * P_{ij} - A*I_o^2 \tag{28}$$

Because of our previous calculations, only the double sum (here denoted U') is calculated for the bounding rectangle and the

corners. For the bounding rectangle, this is:

$$U'_{rect} = \sum_{i=x}^{x+m-1} \sum_{j=y}^{y+n-1} i^2 * P_{ij} \qquad (29)$$

Since P_{ij} is always 1 inside the bounding rectangle, this simplifies to:

$$U'_{rect} = \sum_{i=x}^{x+m-1} i^2 * n \qquad (30)$$

which further simplifies to:

$$U'_{rect} = \frac{m*n}{6} * (6*x*(i+m-1) + (2*m-1)*(m-1)) \qquad (31)$$

Following the same manipulation, U' for one of the top corners (k_0 or k_1) is: (assume corner size is k)

$$U'_t = \frac{k^2*(k+1)^2}{4} + \frac{(k*(k+1)*(k+x))}{2} * (x - \frac{k}{3} - \frac{2}{3}) \qquad (32)$$

Following the same manipulation, U' for one of the bottom corners (k_2 or k_3) is: (assume corner size is k)

$$U'_b = \frac{k^2*(k+1)^2}{4} + \frac{k*(k+1)*(2*k+1)}{3} * (x+m-k-1)+ \qquad (33)$$

$$\frac{k*(k+1)}{2} * (2*k+(k-1)^2 + m*(m+2*x-2*k-2)-2*x+1)$$

Thus, U is:

$$U = U'_{rect} - U'_t(k_0) - U'_t(k_1) - U'_b(k_2) - U'_b(k_3) - A*I_o^2 \qquad (34)$$

The value of W is calculated in exactly the same manner, except

W' is calculated for the left and right corners, rather than the top and bottom:

$$W = W'_{rect} - W'_r(k_0) - W'_l(k_1) - W'_l(k_2) - W'_r(k_3) - A*J_o^2 \qquad (35)$$

The calculation of V is more complex, but the method used is similar to that used above. Recall the formula for V:

$$V = \sum_i \sum_j i*j*P_{ij} - 2*A*I_o*J_o \qquad (36)$$

Since A, I_o, and J_o are already known, we will calculate the value of the double sum (here denoted V'). For the bounding rectangle, the value of V' is:

$$V'_{rect} = \sum_{i=x}^{x+m-1} \sum_{j=y}^{y+n-1} i*j*P_{ij} \qquad (37)$$

Since P_{ij} is always 1 within the bounding rectangle, this simplifies to:

$$V'_{rect} = \sum_{i=x}^{x+m-1} i \sum_{j=y}^{y+n-1} j \qquad (38)$$

By evaluating the sums, this yields:

$$V'_{rect} = \frac{m*n*(m+2*x-1)*(n+2*y-1)}{2} \qquad (39)$$

Unfortunately, each of the corners has to be treated separately. They are (for corner k_0):

$$V'_0 = \sum_{i=x}^{x+k_0-1} \sum_{j=y}^{y+k_0-i+1} i*j \qquad (40)$$

by simplifying, and making the following substitutions:

$$d_1 = x+y+k_0 \qquad d_2 = d_1 - 1$$

$$e_1 = x+k_0 \qquad e_2 = e_1 - 1$$

this becomes:

$$V'_0 = \frac{1}{4}*\left(\frac{e_1^2*e_2^2-x^2*(x-1)^2}{2} - \frac{(d_1+d_2)*e_1*e_2*(2*i+2*k-1)}{3} + \right. \qquad (41)$$

$$\frac{(d_1+d_2)*x*(x-1)*(2*x-1)}{3} + e_1*e_2*(d_1*d_2-y*(y-1)) -$$

$$\left. x*(x-1)*(d_1*d_2-y*(y-1))\right)$$

The value for V'_1 is:

$$V'_1 = \sum_{i=x}^{i+k_1-1} \sum_{j=y+n+i-x-k_1}^{n+y-1} i*j*P_{ij} \qquad (42)$$

By simplifying, and making the following substitutions:

$$d_1 = y+n-i-k_1 \qquad d_2 = d_1-1$$

$$e_1 = x+k_1 \qquad e_2 = e_1-1$$

$$V'_1 = \frac{1}{4}*(((n+y-1)*(n+y)-d_1*d_2) * (e_1*e_2-x*(x-1)) - \qquad (43)$$

$$\frac{e_1^2*e_2^2}{2} + \frac{x^2*(x-1)^2}{2} -$$

$$\frac{(d_1+d_2)*e_1*e_2*(2*x+2*k_1-1)}{3} +$$

$$\frac{(d_1+d_2)*x*(x-1)*(2*x-1)}{3})$$

The value for V'_2 is:

$$V'_2 = \sum_{i=x+m-k_2}^{x+m-1} \sum_{j=y+n-1+x+m-k_2-i}^{y+n-1} i*j*P_{ij} \qquad (44)$$

By simplifying, and making the following substitutions:

$$d_1 = y+n+x+m-k_2-1 \qquad d_2 = d_1-1$$

$$e_1 = y+n \qquad e_2 = e_1-1$$

$$f_1 = x+m-k_2 \qquad f_2 = f_1-1$$

$$g_1 = x+m \qquad g_2 = g_1-1$$

yields:

$$V'_2 = \frac{1}{4}*((e_1*e_2-d_1*d_2)*(g_1*g_2-f_1*f_2) + \qquad (45)$$

$$\frac{(d_1+d_2)*g_1*g_2*(2*x+2*m-1)}{3} -$$

$$\frac{(d_1+d_2)*f_1*f_2*(2*x+2*m-2*k_2-1)}{3} + \frac{f_1^2*f_2^2}{2} - \frac{g_1^2*g_2^2}{2})$$

The remaining corner V'_3 can be specified as follows:

$$V'_3 = \sum_{j=y}^{y+k_3-1} \sum_{i=x+m+j-y-k_3}^{x+m-1} i*j*P_{ij} \qquad (46)$$

By simplifying, making the follow substitutions:

$$d_1 = x+m-j-k_3 \qquad d_2 = d_1-1$$

$$e_1 = y+k_3 \qquad e_2 = e_1-1$$

$$V'_3 = \frac{1}{4}*(((m+x-1)*(m+x) - d_1*d_2)*(e_1*e_2 - y*(y-1)) - \qquad (47)$$

$$\frac{e_1^2*e_2^2}{2} + \frac{y^2*(y-1)^2}{2} -$$

$$\frac{(d_1+d_2)*e_1*e_2*(2*y+2*k_3-1)}{3} +$$

$$\frac{(d_1+d_2)*y*(y-1)*(2*y-1)}{3})$$

And thus the formula for V is:

$$V = V'_{rect} - \sum_{i=0}^{3} V'_i - 2*A*I_o*J_o \qquad (48)$$

And finally, to reiterate, we calculate the angle of the principal axis of inertia, θ, and the maximum and minimum inertia, I_{max} and I_{min}, as:

$$\theta = \frac{1}{2}*\tan^{-1}(\frac{V}{(U-W)}) \qquad (49)$$

$$I_{max} = (U+W) + \sqrt{V^2+(U-W)^2} \qquad (50)$$

$$I_{min} = (U+W) - \sqrt{V^2+(U-W)^2} \qquad (51)$$

8. Bibliography

Barrow [1978]
Barrow, H. G., and Tenenbaum, J. M., Recovering Intrinsic Scene Characteristics from Images, Computer Vision Systems, Hanson, A. R. and Riseman, E. M., eds., Academic Press, New York, 1978.

Baumgart [1974]
Baumgart, B. G., Geometric Modeling for Computer Vision, Stanford Artificial Intelligence Laboratory, Memo AIM-246, October, 1974.

Caplan [1967a]
Caplan, B., Swedish Computerized Mill: Leap in Lumber Technology, Canadian Forest Products, Dec. 1967, pp. 28-32.

Caplan [1967b]
Caplan, B., Sawing by Computer, Grading by X-Ray at Swedish Mill, Canadian Forest Products, May 1967, pp. 44-45.

Castleman [1979]
Castleman, K. R., Digital Image Processing, Prentice-Hall, New Jersey, 1979.

Duda [1973]
Duda, R. O. and Hart, P. E., Pattern Classification and Scene Analysis, John Wiley & Sons, New York, 1973.

Ellinger [1979]
Ellinger, H., Morgan, I. L., Klinksiek, R., and Hopkins, F., Tomographic Analysis of Structural Materials, Proceedings of the Society of Photo-Optical Instrumentation Engineers (SPIE), 182, pp. 179-186, (1979).

Gordon [1975]
Gordon, R., Herman, G. T., and Johnson, S. A., Image Reconstruction form Projections, Scientific American, October, 1975, pp 56-68.

Hwang [1979]
Hwang, J. J., Hall, E. L., Lee, C. C., Nalesnik, W. J., and Archer, C. R., "Global Local Edge Coincidence Segmentation for Computer Tomography Images," Proceedings of the Sixth Conference on Computer Applications in radiology & Computer/Aided Analysis of Radiological Images, IEEE, New York, NY, 1979, pp. 312-320.

IIS [1979]
Product Description, Model 70/F Image Computer and Display Terminal, Internation Imaging Systems, Sunnyvale, CA, 1979.

Henrich [1979]
Henrich, G., Mai, N., Backmund, H., Preprocessing in Computed Tomography Picture Analysis: A "Bone-Deleting" Algorithm, Journal of Computer Assisted Tomography, 3, no. 3, pp. 379-384, June, 1979.

Herman [1979]
Herman, G. T., Image Reconstruction from Projections, Springer-Verlag, New York, 1979.

Hopkins [1981]
Hopkins, F. F., Morgan, I. L., Ellinger, H. D., Klinksiek, R. V., Meyer, G. A., and Thompson, J. N., Industrial Tomography Applications, IEEE Transactions of Nuclear Science, NS-28, no. 2, (1981).

Katz [1978]
Katz, M. B., Questions of Uniqueness and Resolution in Reconstruction from Projections, Springer-Verlag, New York, 1978.

Kruger [1978a]
Kruger, R. P. and Cannon, T. M., The Application of Computed Tomography, Boundary Detection, and Shaded Computer Graphics Reconstruction to Industrial Inspection, Material Evaluation, April, 1978, pp. 75-80.

Kruger [1978b]
Kruger, R. P., Cannon, T. M., Lundy, A. S., and Morris, R. A., Simulation and Preliminary Performance Characteristics of a Computed Tomography Device for Industrial Application, Proceedings of the Society of Photo-Optical Instrumentation Engineers (SPIE), 155, pp. 178-183, (1978).

Larsen [1977]
Larsen, G. N., Glenn, W., Kishore, P. R. S., Davis, K., MacFarland, W., Dwyer, S. J., Computer Processing of CT Images: Advances and Prospects, Neurosurgery, 1, no. 1, (1977).

Marr [1978]
Marr, D., Representing Visual Information, Computer Vision Systems, Hanson, A. R., and Riseman, E. M., eds., Academic Press, New York, 1978.

Marr [1979]
Marr, D. and Hildreth, E., Theory of Edge Detection, A. I. Memo 518, Massachusetts Institute of Technology, 1979.

Miller [1964]
Miller, D. G., Detection of Rot in Wood by Electronic X-Ray Fluoroscopy, British Columbia Lumberman, Oct. 1964.

Nevatia [1979]
Nevatia, R and Babu, K. R., Linear Feature Extraction and Description, Proceedings of the 6th International Joint Conference on Artificial Intelligence, pp. 639-641, (1979).

Nevatia [1981]
Nevatia, R, Price, K., and Vilnrotter, F., Describing Natural Textures, Proceedings of the 7th International Joint Conference on Artificial Intelligence, pp. 649-644, (1981).

Reimers [1980]
Reimers, P., Heidt, H., Stade, J., and Weise, H., Applications of Computerized Axial Tomography to the Field of Non-destructive Testing, Materialpruf (Material Testing - Germany), 22, no. 5, (1980).

Rhodes [1979]
Rhodes, M. L., "Towards Fast Edge Detection for Clinical 3-D Applications of Computer Tomography," Proceedings of the Sixth Conference on Computer Applications in radiology & Computer/Aided Analysis of Radiological Images, IEEE, New York, NY, 1979, pp. 321-327.

Selfridge [1981]
Selfridge, P. G. and Prewitt, J. M. S., Organ Detection in Abdominal Computerized Tomography Scans: Application to the Kidney, Computer Graphics and Image Processing, 15, 265-278 (1981).

Shirai [1975]
Shirai, Y., Analysing Intensity Arrays Using Knowledge about Scenes, The Psychology of Computer Vision, Winston, P. H., ed., McGraw-Hill, New York, 1975.

Szymani [1981]
Szymani, R., and McDonald, K. A., Defect Detection in Lumber: State of the Art, Forest Products Journal, 31, No. 11, pp. 34-44, (Nov. 1981).

Tomita [1979]
Tomita, F., Shirai, Y., and Tsuji, S., Description of

Textures by a Structural Analysis, Proceedings of the 6th International Joint Conference on Artificial Intelligence, pp. 884-889, (1979).

Winston [1981]
Winston,P. H., Horn, B. K. P, LISP, Addison-Wesley, Reading, Massachusetts, 1981.

WWP [1968]
Computerized Lumber Mill Grades by X-Ray, Scales Electronically, Wood and Wood Products, Feb. 1968, pp. 28-31.