01 Jan 1989

Automatic Color Segmentation of Images with Application to Detection of Variegated Coloring in Skin Tumors

Scott E. Umbaugh

Randy Hays Moss
Missouri University of Science and Technology, rhm@mst.edu

William V. Stoecker
Missouri University of Science and Technology, wvs@mst.edu

Recommended Citation

S. E. Umbaugh et al., "Automatic Color Segmentation of Images with Application to Detection of
Variegated Coloring in Skin Tumors," *IEEE Engineering in Medicine and Biology Magazine*, Institute of
Electrical and Electronics Engineers (IEEE), Jan 1989.
The definitive version is available at https://doi.org/10.1109/51.45955

Automatic Color Segmentation of Images with Application to Detection of Variegated Coloring in Skin Tumors

Scott E Umbaugh, Randy H. Moss, and William V. Stoecker
Electrical Engineering Department,
University of Missouri–Rolla (S.E.U., R.H.M.)
Stoecker Moss & Co., University of Missouri–Rolla, and
University of Missouri–Columbia (W.V.S.)

COMPUTER VISION is becoming a useful tool in a wide variety of applications. One major area of accelerated growth is the field of medicine, where research involving automatic identification of structures in CT scans [1], quantification of DNA content [2], and radionucleotide scans [3] shows promise of development of tools to serve as diagnostic adjuncts for medical professionals. This paper describes a computer vision system to serve as the front-end of a medical expert system that will automate visual feature identification for skin tumor evaluation. The general approach is to create different software modules that will detect the presence or absence of critical features. Image analysis with artificial intelligence (AI) techniques, such as the use of heuristics incorporated into image processing algorithms, is the primary approach.

The hardware for this visual subsystem consists of a video camera interfaced to a digitizing board that is part of a microcomputer system. The front-end visual system when completed will provide for automatic recognition of the most predictive features, allowing the expert system software to classify a tumor based on stored information, with minimal user interaction. This classification will be a differential diagnosis, listing the possible diagnoses along with their corresponding probabilities.

For the original expert system, AI/DERM [4], the user was required to enter extensive tumor information—the presence or absence of about 300 features. Clinical tests showed that this user interaction was the weak link in the chain; quite often a misdiagnosis was not caused by errors in the expert system software, but was due to observer error. Many visual features, such as indistinct border, color, and other critical variables were poorly defined. It was found that poorly defined features were not reliably identified by different observers, even by experienced dermatologists.

The major difficulty observed was the lack of consistency in feature identification. With a dermatologist entering the feature information, the identification of specific features was biased by the global information that was part of his knowledge domain. For example, if he suspected that the tumor was malignant melanoma, and he knew that this classification generally implied irregular coloring, then variegated coloring would be marked as a positive on the feature entry form, even for cases in which coloring was only mottled and not variegated. Such bias made it very difficult to define some features, as the expert was using higher level information to redefine each feature. By automating the feature identification aspect of the diagnosis, a more consistent paradigm can be developed for the entire diagnostic procedure. Also, the specific features will be better defined, thus providing a complete and consistent model for the classification of skin tumors.

The results reported in this paper include part of the software required to implement this system. On a broad scale, this research addressed the problem of segmentation of a digital image based on color information. As an application example, a feature called variegated coloring was selected for identification. This feature has high predictivity for diagnosis of malignant melanoma—the fastest growing and deadliest skin cancer. Variegated coloring is defined as a tumor with two or more colors within the tumor border. Furthermore, these colors completely partition the tumor after other features, such as ulcer and and crust, are masked out.

There were two main reasons that the broader problem of using color information to segment a digital image was attacked initially. First, segmenting the image into color objects provides more information than simply identifying a specific feature, and, second, the features themselves are not precisely defined. For example, one dermatologist's definition of variegated coloring may not be the same as another's, and the simplistic definition given above was continually being changed during this research. Thus, it was decided that the automatic segmentation of the image by color information would lead to the most useful results.

The algorithm that was developed to segment the image based strictly on color information was shown to be a useful aid in the identification of tumor border, ulcer, and other features of interest. As a specific application example, the method was applied to 200 digitized skin tumor images to identify the feature called variegated coloring.

BACKGROUND INFORMATION

The literature reviewed for this research consisted primarily of two main categories: color spaces and transforms, and image segmentation techniques. The information discussed in the literature was used as background material for the development of the algorithm presented later in this paper.

Color Spaces and Transforms. A color space is a geometrical and mathematical representation of color. Most of the spaces reviewed here attempt to relate the way in which they are defined to the way that humans perceive color. The search for a useful metric to distinguish perceptual color differences has been pursued since the days of Isaac Newton [5]. There is no general method that is applicable to all domains; the number of variables involved make for complexity such that a complete theoretical analysis is not feasible in most practical applications.

In any problem of color quantification, the first step toward a solution is to define the color space. Historically, many

different representations have been defined, but each was developed for a specific purpose. Some are applicable to the artist's concept of color, usually pertaining to pigment mixtures, and represent a subtractive model of color primitives. The additive model is normally used when modeling light and color, or human perception of color [5].

An Historical Perspective. Artists and philosophers have contemplated the nature of the color experience at least since Aristotle in 350 B.C. [5]. Aristotle believed that colors were composed of mixtures of black and white. His ordering of colors was based on his observance of the colors of the rainbow, believing that they were ordered by brightness. Today, it is known that this ordering is based on the wavelengths of light.

The scientific approach to light and color began with Newton's treatise, *Opticks*, published in 1704. The history can be traced from Newton through Young, Helmholtz, Maxwell, and Mach [5]. Maxwell conceptualized a system that is the basis for many different aspects of color research today. He proposed three imaginary primaries that would be combined to produce any color [5]. He modeled this as an equilateral triangle with each of the vertices representing one of the primary colors. In the exact center of this triangle is the color "white", which consists of equal parts of the three primaries (see Fig. 1). This space is sometimes referred to as Maxwell's triangle [6].

The Munsell system [5, 6], developed in 1941, is often used in psychological research and provides a good basis for describing how humans perceive color. Munsell identifies colors in terms of three attributes: hue, value, and chroma. Hue is defined by what is normally identified as "color"; there are five principal hues in Munsell color space—red, yellow, green, blue, and purple. Referring back to Maxwell's triangle, if a circle is circumscribed within the triangle, hue can be defined by an angle within this circle measured from an arbitrary reference point. Chroma can be defined as the distance from the white point in the center; it is a measure of how pure the color is. Value corresponds to another dimension, which can be modeled mathematically by a constant multiplier—i.e., value is a measure of brightness (see Fig. 2).

Other color spaces that are offshoots of the Munsell space have been defined. The IHS space (Intensity/Hue/Saturation) is quantified in [7]. This color space is more useful to the engineer than the Munsell system, as it can be modeled mathematically in a reasonable form. IHS space is closely correlated to the conceptual model given in Fig. 2. Intensity is simply the sum of the magnitude of the three basis vectors—normally red, green, and blue. It most closely correlates to brightness. Hue is approximately proportional to the mean wavelength, and saturation measures the purity of the color. A very saturated color, or pure color, can be desaturated by adding white to it; for example, pink is a desaturated red.
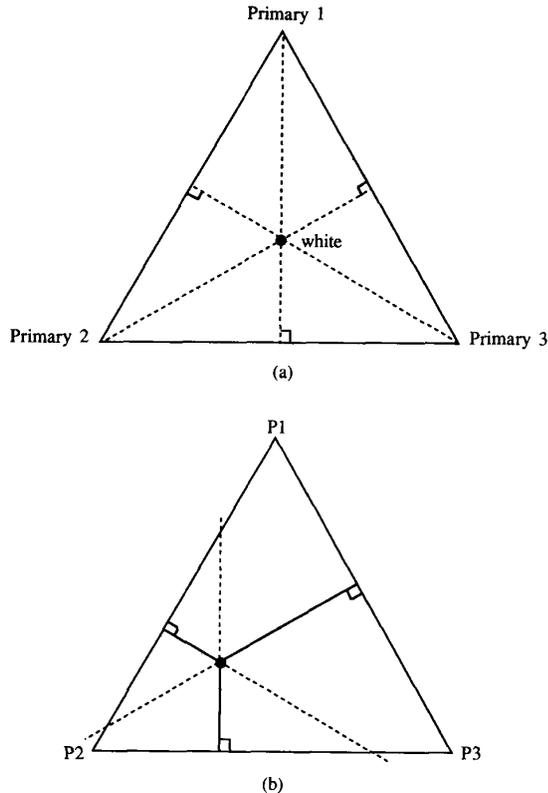

(a)

(b)

Figure 1. a) Maxwell's Triangle. White is at the center of the triangle. The three primary colors are at the vertices of the triangle. b) An arbitrary color, pictured as a point within the triangle, is defined by its components of the three primaries. These components are obtained by drawing a line perpendicular to the side of the triangle opposite the primary's vertex through the color point. This figure shows the P1, P2, and P3 components as solid lines of the color point.
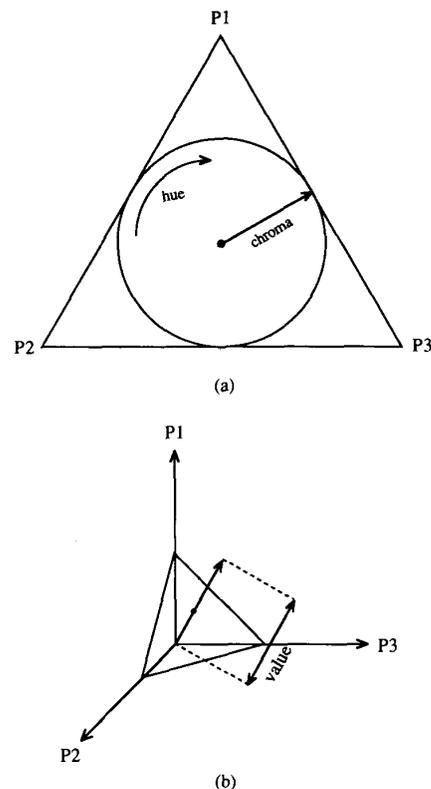

(a)

(b)

Figure 2. a) A conceptual model of the Munsell Color System superimposed on Maxwell's triangle. Hue is measured by an angle from an arbitrary reference point, and chroma is measured by the distance from the white point to the color point. b) Value is measured as the length of the color vector from the origin. Here it is shown as the vector length in an orthogonal 3-D space.

Conceptually, the Munsell system, or a quasi-Munsell space such as IHS, provides a good model for human perception of color.

Human Perception of Color. There are three different types of sensors in the eye for the perception of color. These photoreceptors are called cones, and each of the three types has a different visual pigment that will respond to light (photons); one pigment has maximum sensitivity to light with a wavelength of 455 nm (blue), another 535 nm (green), and the third 570 nm (red) [8]. Approximate spectral response curves, or transfer functions, of the cones have been empirically determined by color matching experiments (see Fig. 3) [6]. (For an explanation regarding negative values of the curves see [6].) These curves are called the tristimulus curves, and it is known that the actual transfer functions of the cones are linear transformations of these curves; however, the exact coefficients are not known [9].

These curves show that the blue photoreceptor is spectrally separate from the other two, which have their maxi-
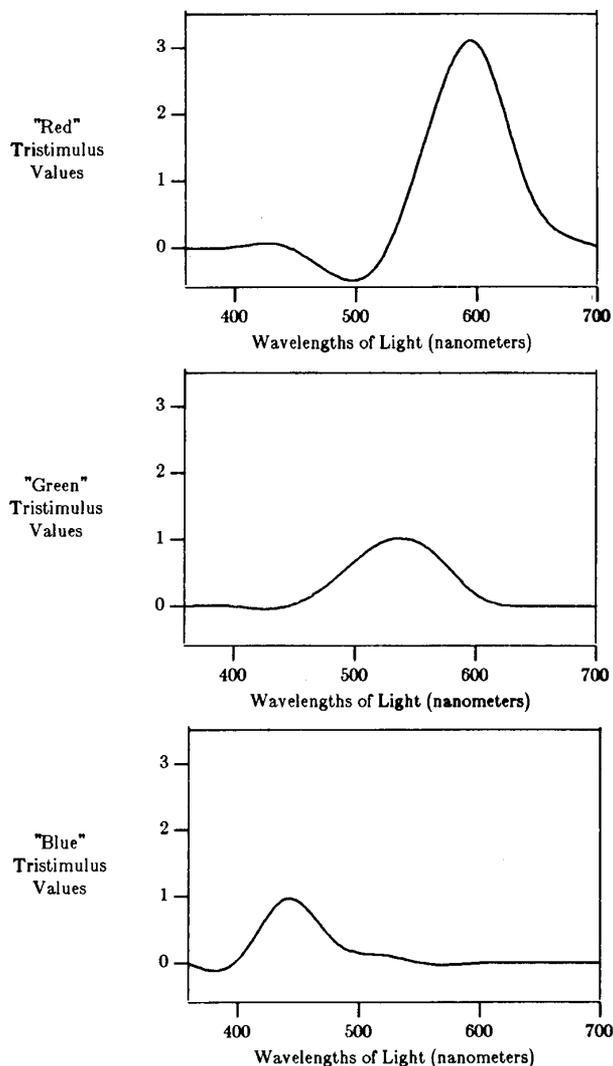

Figure 3. Tristimulus Curves. These curves were approximated from data contained in the CIE 1964 Supplementary Standard Colorimetric System Tables [6].

mum sensitivities close together near the yellow part of the spectrum. This separation means that the blue cones have a wider range of spectral contrast [8]—i.e., colors near each other in the blue zone of the color space are more readily distinguished by human vision than colors that are equally close, but near the red or green zones.

There are three major types of theories, with their corresponding models, that describe the underlying mechanisms that comprise color vision. These are: 1) component theories, 2) opponent theories, and 3) zone theories [5]. Each of these theories proposes a different model for human color vision.

There is still much controversy over which is the "best" model for the human vision system. It appears that it is largely domain dependent, and the "best" model is the one that facilitates the solution of the problem being examined. The issue is very complex—not only are the physiological aspects important, but psychological ones as well.

Colorimetry. Colorimetry is the science of defining a mathematical model of color sensation so that a measurement can be derived for distinguishing between different colors, and being able to measure the degree to which the colors differ [6]. It involves quantifying the components that comprise the color, and being able to measure them in such a way that when viewed by a normal observer, under the same conditions as when measured, the observer will experience a metameric match [5]. Metameric refers to two colors that appear the same to the human visual system, even though the actual spectral composition of the incident light is different.

The basic concepts of colorimetry provide a foundation for the CIE (*Commission Internationale de l'Eclairage*) System, which is the standard system used throughout the world today for quantifying color. (A complete discussion of this system is given in [6].) One of the most basic concepts involved here is that of trichromatic generalization. This concept states that given three primary colors, or wavelengths of light, many colors can be matched by the additive mixing of these three primaries by individually adjusting the intensity of each. It is this premise that facilitates the digitization of color images by the use of a monochrome camera and three color filters, with the resulting information from each filter used to display the image on a color (RGB) monitor.

Linear Transforms. Many linear transforms of color information are defined in the literature. Most of them assume the initial vectors to be three-dimensional, with the standard basis being red, green, and blue components, hereafter referred to as RGB. The transformed vectors are then obtained by multiplying the RGB vector by a 3 × 3 transform matrix. Two advantages of linear versus nonlinear transforms are efficiency of computation and lack of singularities [10].

Due to the singularities inherent in most nonlinear transforms, they do not lead to a consistent definition of color throughout the color space. On the other hand, a linear transform will allow a consistent definition, provided the transformation matrix is well-conditioned [10]. The inconsistencies that occur in nonlinear transforms will cause problems if not dealt with appropriately. If, however, the domain of interest does not include the subspaces of the space that contains the singularities, then the difficulties associated with nonlinear transforms are of no significance, as was the case with the algorithm developed for this project.

Nonlinear Transforms. The tristimulus curves [6, 9] indicate the nonlinearity inherent in the human visual system. Therefore, when an application requires the simulation of human visual response, it is appropriate to explore nonlinear transforms. One of the standard nonlinear transforms produces what are called chromaticity coordinates [7]. These are

defined as follows:

$$r = R/(R + G + B),$$

$$g = G/(R + G + B), \text{ and}$$

$$b = B/(R + G + B).$$

When plotted as a function of two of these coordinates, the result is called a chromaticity diagram [6]. The CIE colorimetric system, referred to earlier, is based on this concept. This system maps tristimulus values onto chromaticity coordinates [6], thus describing a system that will have a fair correlation to the human visual response. Many other nonlinear transforms are defined in the literature (see [6]). Quite often, in image processing, a color space is defined to facilitate image segmentation so that a useful representation on a level higher than that of the pixel can be derived.

Image Segmentation. Image segmentation is important in many computer vision and image processing applications. Division of the image into regions corresponding to objects of interest is necessary before any processing can be done at a level higher than that of the pixel. Identification of real objects, pseudo-objects, shadows, or actually finding anything of interest within the image requires some form of segmentation.

For example, identification of shadows may help determine the three-dimensional shape of an object, or, if the object is known, the orientation of the object in space. Other examples include using satellite data to identify urban areas [11], specific vegetation [11], or even insect infestation within a certain type of tree [12].

Conceptually, image segmentation methods will look for objects that either 1) have some measure of homogeneity within themselves, or 2) have some measure of contrast with the objects on their borders [11]. Most image segmentation algorithms that have been used are modifications, extensions, or combinations of these two basic concepts.

Some of the problems associated with image segmentation are due to noise in the image and digitization of a continuous image. This noise is caused by digitization, the camera, the lenses, the lighting, etc., and can be dealt with through the use of statistical methods [11]. Spatial digitization can cause problems regarding connectivity of objects [9]. These problems can be resolved with careful connectivity definitions and heuristics applicable to the specific domain.

The following review divides image segmentation techniques into three main groups: 1) region growing, 2) clustering methods, and 3) edge detection (see Fig. 4). Clustering methods are separated from region growing methods because, in this context, region growing methods are restricted to methods that primarily use the spatial domain, the two-dimensional row and column image space, while the clustering techniques could be applied to any domain (spatial domain, color space, feature space, etc.).

Region Growing. Region growing refers to a class of image segmentation methods where the goal is to find regions that represent objects or meaningful parts of objects. The method is based primarily on spatial considerations. Some of the techniques used are local, in which small areas of the image are processed; others are global, where the entire image is considered during processing [7]. Methods that can combine local and global techniques, such as split and merge [13], are referred to as state space techniques and use graph structures to represent the regions and their edges (boundaries) [7].

Various split and merge algorithms have been described [13, 14], but they all are most effective when heuristics
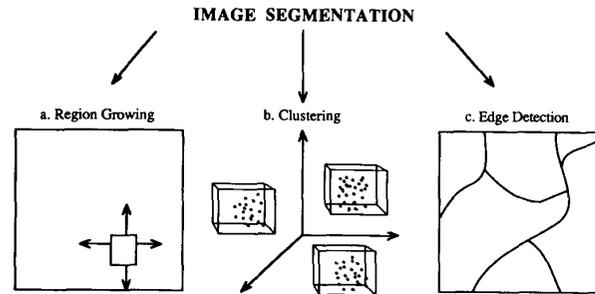
IMAGE SEGMENTATION

Figure 4. Image Segmentation techniques are divided into three major categories. a) Region growing is performed within the image by finding homogeneous regions and growing them until they no longer meet the homogeneity criteria. b) Clustering techniques look for groups, or clusters, in domains other than the spatial domain of the image. Here clusters are found in a 3-D space, e.g., RGB space. c) Edge detection methods look for edges, boundaries, or lines usually via a differentiation operator.

applicable to the domain under consideration can be applied. This gives a starting point for the initial split. In general, the split and merge technique proceeds as follows. First, the image is split into equal sized regions, then some type of statistical method is applied to determine a measure of similarity within each of the regions. This measure may include texture variation, color variation, intensity variation, or other features of interest. Once this information has been calculated for each of the regions, the image is ready for the next step in processing.

The next step will normally be a homogeneity test for each of the regions. The regions that are acceptable by the criteria of the homogeneity test will be left alone; the regions that do not pass the test will be split into more subregions. After the split is made, a merge is attempted. This merge procedure will attempt to merge each region with its neighboring regions, or sub-regions, and if the resulting region is acceptable to the homogeneity test, these regions will be merged. This procedure stops when all regions that have been formed pass the homogeneity test.

Other region growing techniques are conceptually similar to the techniques described here. A more complete discussion of region growing techniques can be found in [7].

Clustering Techniques. Clustering techniques are image segmentation methods whereby individual elements are placed into groups; these groups are based on some measure of similarity within the group. The major difference between these techniques and the region growing techniques is that domains other than the spatial domain may be considered as the primary space being used for the segmentation.

One method of image segmentation based on clustering, in widespread use, is the method of taking the space of interest and splitting the space into regions by setting limits on each of the dimensions for each separate region [15, 16]. In the case of using an RGB color space, this would mean taking the three-dimensional color space and dividing it into rectangular parallelepipeds with edges parallel to the axes in the color space [16].

Recursive region splitting [14] is a clustering method that has become a standard technique. This method uses a thresholding-of-histograms technique to segment the image. A set of histograms is calculated for a specific set of features, then each of these histograms is searched for distinct peaks. The best peak is selected and the image is split into regions based on this thresholding of the histogram.

Many methods may actually be a combination of region

growing methods and clustering methods. The segmentation method of dividing the image based on clusters in color space [14, 17] can be implemented as strictly a clustering method, or spatial considerations may also be included. Optimal image segmentation will most likely be achieved with the combined approach.

Edge Detection. Edge detection, as a method of image segmentation, is performed by finding the boundaries between objects, thus defining the objects themselves, indirectly. This method is usually implemented by first marking points that may be a part of an edge. These points are merged into line segments, and the line segments are then merged into object boundaries [7].

The most common method of finding edges in a digitized image is to apply a spatial differentiation operator to small blocks of pixels, local neighborhoods, within the digitized image [18]. Places in the image where first order differentiation returns a large number mark points of rapid change, thus indicating the possibility of an edge. These edge points represent local discontinuities in a specific feature, such as brightness, color, or texture [18]. Many edge detection operators have been defined, but most are based on these fundamental concepts.

One of these operators, the Laplacian operator [9], is used in the Marr-Hildreth algorithm [19], which has been used in other areas of this research project [20]. This algorithm provides a good model for the mammalian vision system [19]. The Marr-Hildreth algorithm uses a Gaussian smoothing filter to eliminate small discontinuities such as hairs, and then applies the Laplacian to find potential edge points [20].

Quite often, in edge detection, heuristics applicable to the specific domain must be employed in order to find the true object boundaries (such as the filtering out of hairs mentioned above), and the process of finding these boundaries is no easy task. Perhaps finding edges of different features and applying AI techniques at a higher level to correlate the feature edges found to the specific domain will give the best results. (For a complete discussion of this concept see [15] and [21].) For a more complete list of references regarding edge detection see [9].

MATERIALS AND METHODS

The images used in this research were digitized with a monochrome video camera (Ikegami Model ITC-48) interfaced to a Gould DeAnza Image Processing System Model IP8400 and a Digital Equipment Corporation VAX 11/780 minicomputer. They were digitized from 35mm color photographic slides obtained from a private dermatology practice and from New York University or, in one case, from a pamphlet obtained from the American Cancer Society. The digital images had a spatial resolution of 512 × 512 pixels and a grey scale resolution of 8 bits (256 levels). The color images were obtained by digitizing the image three times, each time using a different filter for each of the red, green, and blue planes.

The software developed for this research was written in the C programming language on the VAX minicomputer. The color space used was a nonlinear transform of rectangular RGB space. This transform led to a two-dimensional color space and a one-dimensional brightness space. For this research, the brightness space was ignored, and only the two-dimensional color space was used. The motivation for this was to avoid splitting color objects that were partially in shadow into separate objects. This was of particular importance in the identification of tumor borders in three-dimensional tumors. Further, as a method of color space segmentation, this color space was quantized by setting ranges on each of the two dimensions.

As the target system for this development is a microprocessor-based system, there was a necessity to reduce the amount of data (each image is 768 kbytes) to be processed, and to make the processing algorithm as efficient as possible. This compression was accomplished by such methods as color quantization to reduce color information, averaging to reduce spatial data, and a generating code that was as efficient as possible.

The algorithm itself consisted primarily of six steps, as follows:

1. averaging to reduce spatial data and eliminate noise,
2. masking out of features such as ulcer and crust,
3. splitting the color space into different colors,
4. filtering the results to aid in segmenting the image into color objects,
5. labeling and finding the area of each color object,
6. higher-level processing to determine the final decision regarding the feature identification.

It should be noted that steps 2 and 6 are dependent on the specific feature under consideration.

The Color Space. The original color space (RGB) was defined with red, green, and blue as orthogonal axes, with the corresponding basis vectors being unit vectors along each axis. To facilitate color quantization, spherical coordinates were chosen as a basis for this space (see Fig. 5). The transforms from RGB to spherical coordinates are as follows [22]:

$$L = \sqrt{R^2 + G^2 + B^2}$$

$$\text{Angle A} = \cos^{-1}\left[\frac{B}{L}\right]$$

$$\text{Angle B} = \cos^{-1}\left[\frac{R}{L \times (\sin\,(\text{Angle A}))}\right]$$

where (R, G, B) is the triple corresponding to the values for red, green, and blue in the original RGB space. This transformation splits the color space into a two-dimensional color space represented by the two angles, Angle A and Angle B, and a one-dimensional intensity (brightness) space represented by the vector length L.

If an equilateral triangle is superimposed on the color space (see Fig. 6), with the vertices of the triangle at the points (1,
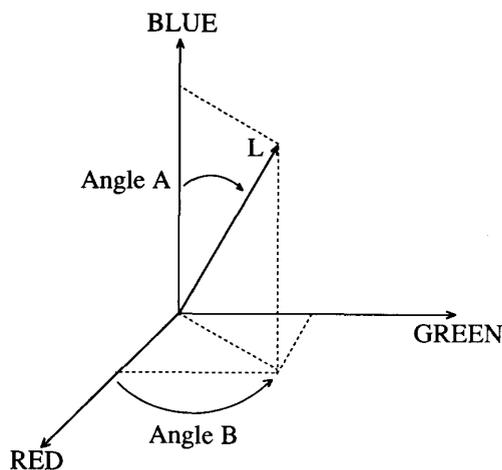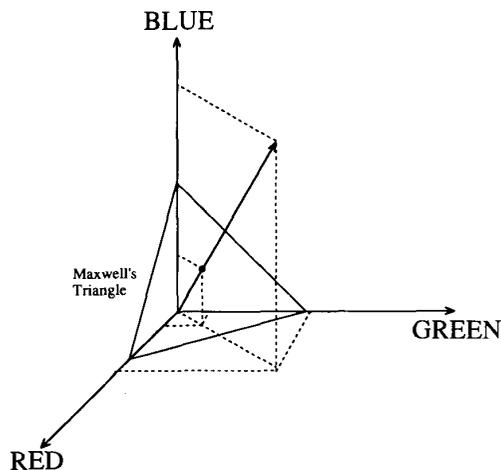

Figure 5. The Color Space.

Figure 6. The Color Triangle.

0, 0), (0, 1, 0), and (0, 0, 1), or any other points of equal distance on each axis, then the point where a vector intersects this triangle corresponds to the color [6]. This point is described by the two angles, A and B, which define a color quantization concept.

Color quantization reduces the amount of data that needs to be processed. The two angles are used to define a space that is then quantized with the idea of reducing this two-dimensional angle space to a one-dimensional color space.

The Color Segmentation Algorithm. The C programming language was chosen to implement the algorithm due to its ease of transportability, since the target system is a microprocessor-based computer. Each step in the processing chain was implemented by a separate program. These programs communicated by writing their outputs to disk files, which were then read as inputs to the next program in the chain. This method proved to be efficacious for development, as it allowed intermediate results to be grown into full-size images and viewed. Intermediate results could then be used to determine what should be done next or, in some cases, what modifications should be made to the existing programs.

Color Averaging. The first step in the processing chain was color averaging. This was accomplished by reducing the original 512 × 512 × 3 byte image by arithmetic averaging, equivalent to a reduction in spatial resolution. An arithmetic average was performed on a spatial area of 8 × 8 pixels, with each of the three color planes—red, green, and blue—being individuallly processed. This averaging served a two-fold purpose: it reduced the amount of data to be processed by later programs in the chain and it reduced any noise present in the image.

The sources of noise in digitized images are many and varied, including noise caused by digitizing a continuous image, imperfections in the camera and lenses, and imperfections caused by lighting. Many of these noise sources can be modeled as sample functions of a zero-mean Gaussian noise process [9, 11]. Averaging multiple samples will reduce the variance, thus giving a result with less "noise" [9].

The 8 × 8 pixel block used reduced the original image file from 768 kilobytes to 12 kilobytes, reducing processing time for software further down the chain, thus allowing this application to run efficiently on a microprocessor-based computer system.

Feature Masking. In addition to the digitized images, a database of feature information was created by a dermatologist using software developed by the research team. This

software allows the user to display an image and mark certain blocks (in this case 32 × 32 pixel blocks) as containing a specific feature. Through the use of these feature-files, specific sections of an image can be selected for processing by masking out blocks within the image that are not of interest.

Feature masking was important in the development of the feature identification software modules in this project. With the feature marking software developed, the research team was able to proceed independently on each module. The feature files provided a data base that could be used to provide the necessary information to a module under development.

Color Space Segmentation. The color space segmentation algorithm is outlined as follows:

1. Convert the (R, G, B) triple into spherical coordinates— (L, Angle A, Angle B).
2. Find the minima and maxima of Angle A and Angle B.
3. Divide the subspace, defined by the maxima and minima, into equal-sized blocks, with size based on the symbolic name NUM_OF_COLORS.
4. Calculate the means and variances (see [23]) for R, G, B, L, Angle A, and Angle B. These, along with the maxima and minima and number of vectors in each color, are then printed to the standard I/O device (normally the terminal from which the program was running).
5. Using the color numbers (e.g., if the color space was split into 4 colors, then the color numbers would be 1, 2, 3, 4), create an image file with the number corresponding to the color into which the pixel or group of pixels fell when the color space was segmented.
6. Using the means of (R, G, B) for each separate color, create an image file with each color vector replaced by the means of (R, G, B) of the color into which the corresponding pixel or group of pixels fell in the color space segmentation.

The output file used in further processing is the single-plane file that consists of color numbers only—this corresponds to a monochrome image file.

Although the color file was not used in further processing, it was useful for human observation during development as an aid in determination of success up to this point. The statistics, means and variances, were not used in the final results presented here, but may be used in higher level processing.

Object Filtering. The purposes of object filtering were: to filter out small objects that would not be identified as color objects by an expert, and to fill in holes and round objects out, as would the visual system of an expert dermatologist observing a skin tumor.

As this was a preprocessing step to object labeling, a consistent definition of connectivity was needed. Connectivity refers to those neighboring pixels that are considered to be connected to a given pixel. Eight-connectivity means that all eight pixels surrounding a given pixel are connected to it; four-connectivity means that the given pixel is connected only to the pixels directly above and below and to the right and left, the edge neighbors [9]. Six-connectivity includes the edge neighbors and two of the diagonal neighbors.

The definition chosen was that of six connectivity. This choice solves the connectivity paradox and satisfies the Jordan curve theorem [9]. Using six-connectivity also avoids the problem of having to differentiate between object and background, which would be the case if eight-connectivity and four-connectivity were used for object and background, respectively. The six-connectivity model used contained the upper left and lower right corners in addition to the four edge

neighbors. Although this definition was biased towards objects in one diagonal direction, this was not a problem since relatively large, irregular shaped objects were of interest.

Using six-connectivity to define neighboring blocks, the filter searches through the image, block by block, looking for blocks that have neighbors that are different from the given block. If a block is found that has four or more neighbors that are the same, that block is then replaced with the value of the ones that are the same. This procedure has the effect of eliminating isolated blocks, filling in holes, and removing single blocks protruding from objects, thus rounding-off objects. Once this filtering is complete, the next step is to label objects.

Object Labeling. A modified version of a binary sequential labeling algorithm was used [9, 24]. This algorithm did not require recursive calls, thus allowing sequential scanning of the image. Thus, when a pixel was being processed, it was known that its neighbors to the left and above had already been labeled.

Higher Level Processing. This refers to processing that will be done by the feature identification module. This part of the software is beyond the scope of this paper. However, a simple two-step algorithm was developed for the identification of a specific feature, variegated coloring, so that some preliminary results could be obtained and the feasibility of the methodology could be demonstrated.

DISCUSSION AND RESULTS

When the algorithm was being developed, many different approaches were attempted. This section will explore the work and experimentation that led up to the development of the algorithm described above.

Experimentation. The original approach was to use an image segmentation technique based on histogram thresholding, a method that has been used effectively in various applications [13, 14, 18, 25] and seemed to be the logical first step. For this approach to work, the histograms must have distinct peaks (see Fig. 7). In addition, the histograms must exhibit more than one peak, so that a reasonable threshold can be found that can be used to split the image into partitions.

After much experimentation, it was found that this method would not work for digitized skin tumor images. The major problem with this procedure in RGB space was that all of the histograms examined had only one peak. (The images that were selected for this experimentation had a large variance in

color space and thus were thought to be most easily identifiable by computer). Transforming the images into IHS (Intensity, Hue, Saturation) space, also resulted in histograms that were mostly unimodal; the exceptions were not useful for other reasons (see [24]).

The Color Space. The vertices of the color triangle bear some correlation to the human visual system. The placement of blue at the top of the triangle, and the way in which the spherical transform was defined, relates to the physiological fact that the cones in the human eye that see blue are more discriminatory than the red or green sensitive cones [8]. As Angle A varies, while Angle B has a fixed range, for example 40 to 50 degrees, the closer to the blue axis (axis in 3-D space, or vertex in 2-D space) the smaller the area defined within this space. This means that for a region defined by a range of minima and maxima, on Angle A and Angle B, the side of the region that is closest to the blue vertex is shorter than the side that is closest to the line that joins the red and green vertices.

Also, the distortion caused by the transform facilitates the perception based aspect of the image segmentation; the closer to the perimeter of the triangle, the larger the region that is defined by a fixed angle range (see Fig. 8). This is analogous to the observation that as the white point is approached in the color space, a greater number of hues will be observable in a fixed area by the human visual system than on the perimeter of the color triangle. It should be noted that in all of the skin tumor images, the range of color vectors within a skin tumor image were between the white point and the red vertex—if color vectors near the blue vertex were in the image, the color quantization scheme would probably need to be modified.

Results of Color Space Segmentation. The best visual system available today, in terms of pattern recognition, image analysis, and image understanding, is the human visual system. Therefore, a few select skin tumor images and results are presented to demonstrate the success of the segmentation method presented in this paper.

In Fig. 9, a tumor image is shown that exhibits variegated coloring. Fig. 10 shows that image after averaging over an 8 × 8 pixel block and then splitting the color space into four colors, each represented by a different gray level. In addition to demonstrating the success of the segmentation method in
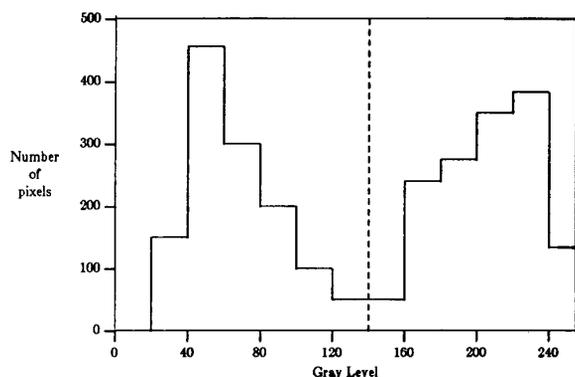

Figure 7. A Gray Level Histogram. This shows the histogram for an 8-bit (256 levels) gray level image. This particular image has a bimodal distribution, so histogram thresholding is performed by dividing the image into two parts. The threshold is illustrated by the dotted line— at about a gray level of 140.
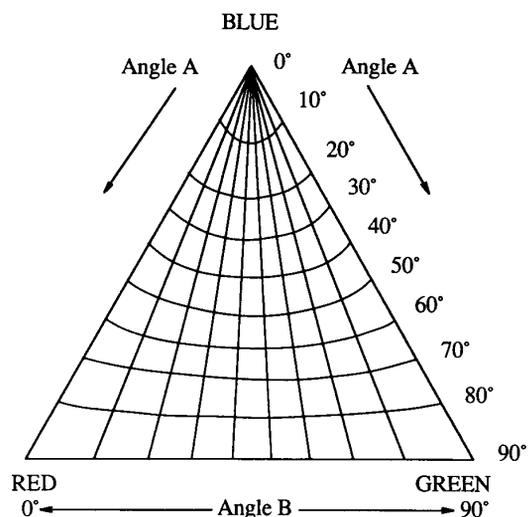

Figure 8. The Color Triangle showing regions defined by 10 degree increments on Angle A and Angle B.
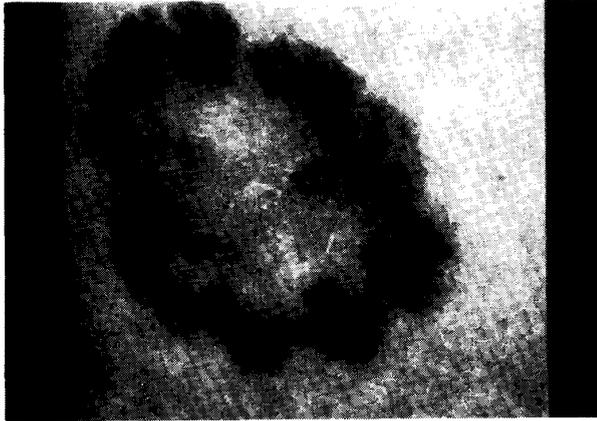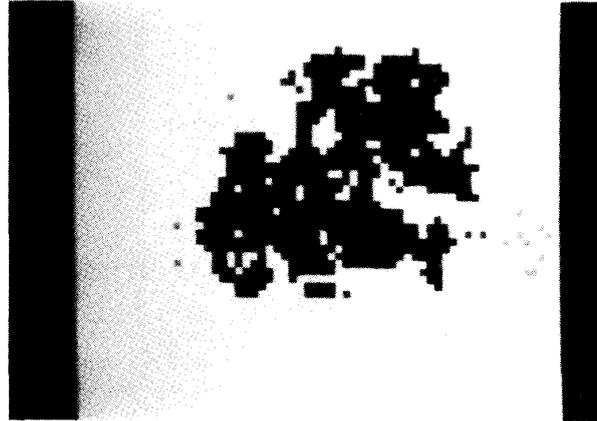
Figure 9. Tumor Image 50, the original.

Figure 12. Tumor Image 13, after color space segmentation.
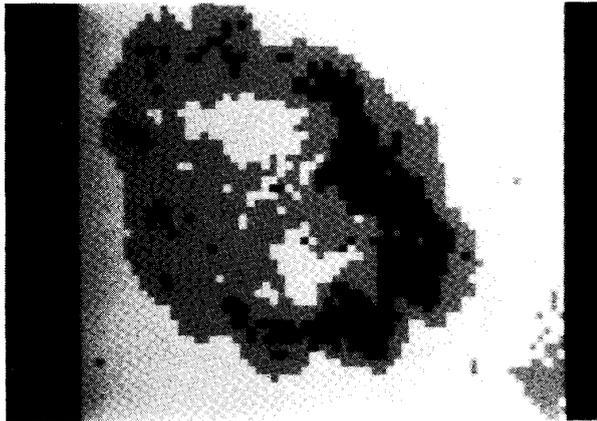

Figure 10. Tumor Image 50, after color space segmentation.

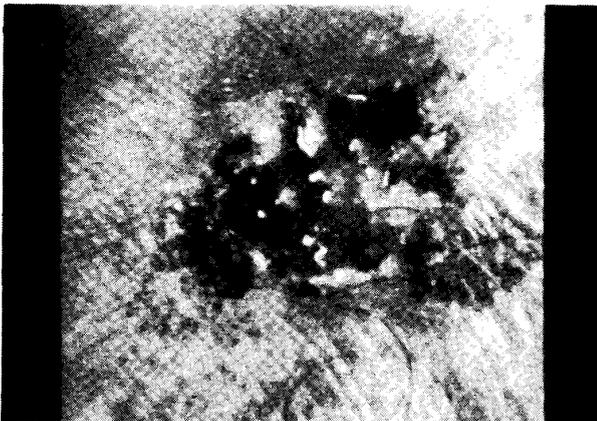Figure 13. Tumor Image 30, the original.
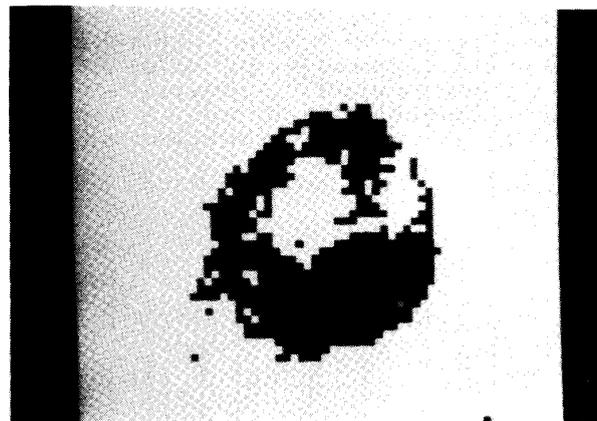

Figure 11. Tumor Image 13, the original.

Figure 14. Tumor Image 30, after color space segmentation.

the identification of variegated coloring, it can be seen that the resulting image would be useful in finding the tumor border.

Figures 11 and 12 are an original skin tumor image and the resulting image after color space segmentation into four colors. This particular tumor exhibits a large ulcerated area which appears darker than the rest of the tumor and has a shiny look. The resulting image shows that the ulcer, represented by the darkest gray level, is accurately separated from the rest of the image. Also, the reflections within the tumor border are separated from the rest of the tumor—they appear the same color as the skin, represented by the lightest gray level in Fig. 12. The medium gray level is actually the pink color of the tumor, and it can be seen that gray level alone would not separate this color from the skin color (see Fig. 11). In Fig. 12, the pink color of the tumor is separated

from the skin color.

Figures 13 and 14 show another tumor image pair before and after color space segmentation into four colors. This tumor exhibits reflections, appearing white in the images, and the ulcer, appearing dark. The resulting image demonstrates that color segmentation is useful in the identification of these features, in addition to assisting in finding the tumor border.

This tumor also illustrates the reason that gray level alone (monochrome processing) will not be successful in finding these features. The left side of the tumor in Fig. 13 appears to be about the same gray level as the ulcer, but it can be seen in Fig. 14 that the two colors are distinctly different—the left side of the tumor is actually a pink color, appearing a moderate gray in this monochrome image, whereas the ulcer is red, appearing black in this image.

Results of the Variegated Coloring Algorithm. The color segmentation algorithm, as applied to the specific feature variegated coloring, was tested on 200 images. As described above, the color segmentation algorithm consists of six major steps: 1) color averaging, 2) feature masking, 3) color space segmentation, 4) object filtering, 5) object labeling, and 6) higher level processing. For identification of variegated coloring, the features masked out (step 2) are ulcer, crust, scaled crust, shiny, and non-tumor areas.

For the purposes of this research, a simple higher level processing algorithm, step 6, was developed that would allow identification of variegated coloring with the information that was obtained by averaging, masking, color splitting, filtering, and labeling the image file. The algorithm consisted of two steps:

1) If the ratio of the area of the tumor, excluding ulcer, crust, and shiny areas, to the entire tumor area was less than 0.5, then variegated coloring was absent.
2) If the number of objects that are greater than 10 blocks, approximately 2 mm^2, consisted of two or more colors, then variegated coloring was present.

This algorithm was chosen due to its simplicity and the fact that it worked effectively on our training set of images. The dermatologist concurred that this was reasonable and was similar to the way in which the identification may eventually be implemented in the expert system.

The color segmentation algorithm was originally applied to 10 images; these images served as the training set for determination of the higher level processing algorithm alluded to above. Of this training set, four exhibited variegated coloring and six did not. The higher level processing algorithm was then defined so that these ten training images would be correctly identified.

Next, the algorithm was applied to 200 test images. The results are displayed in Table I. Of these 200 tumor images, 40 were identified by the dermatologist as possibly exhibiting variegated coloring, so these 40 were not included in the resulting data as it could not be explicitly determined whether they exhibited this feature or not. Eventually, these borderline cases may be used to tune the final higher level processing algorithm that will be implemented. Of the remaining 160

TABLE I
Results of Variegated Coloring Algorithm

Number Correctly Identified	116
Number Incorrectly Identified	44
True Positives	25
True Negatives	91
False Positives	28
False Negatives	16

digitized skin tumor images, 73 percent were correctly identified with variegated coloring being present/absent. It is expected that this percentage will increase as more knowledge is gained and the size of the training set is increased. The next iteration of a higher level processing algorithm will be developed using information gained from the application of the software to the 200 test images, which will effectively become the training set for the next version.

CONCLUSIONS

This systematic approach, allowing feedback from human visual systems at each step in the processing chain, facilitates algorithmic development. The creation of a database that allows independent development of software modules enhances the manageability of a large software project. Also, the concept, borrowed from statistical methods, of using training sets to design the algorithms and test sets to generate results provides for an unbiased observation of the success of the algorithms and when used iteratively, allows for the incremental improvement of the algorithms.

Specifically, this methodology was applied to the creation of software to segment digitized skin tumor images based on color information. It was shown that this color segmentation algorithm may be successful as an aid in finding ulcer, reflections, and tumor border. These results demonstrated the success of image segmentation by color using a spherical coordinate transform of rectangular RGB space.

As a specific example of this technique, an algorithm was presented for identifying variegated coloring in digitized images of skin tumors. The moderate success achieved is indicative of the training/test set paradigm, with a relatively small initial training set, and it is believed that as the training set becomes larger the success ratio will improve.

The software developed to implement this segmentation technique will be part of an expert system, an automatic skin tumor diagnosis system, that will be the application software in a microprocessor-based computer that also contains a visual front-end video camera and digitizing board. The total system has both research and diagnostic potentials.

REFERENCES
1. **Newell JA, Sokolowska E:** Model based recognition of CT scan images. *Proceedings Medinfo 86*, Salamon R, Blum R, Jorgensen M (Eds), Elsevier North Holland, 5 (2):619-623, 1986.
2. **Yasnoff WA, Nayer DA, Mahran HE, Bacus SS, Bacus JW, Marder RJ:** An image analysis system for quantification of cellular DNA content: CAS-100/QDA. *Proceedings Medinfo 86*, Salamon R, Blum B, Jorgensen M (Eds), Elsevier North Holland, 5 (2):690-692, 1986.
3. **Toga AW, Arnicar-Sulze TL:** Digital image reconstruction for the study of brain structure and function. *Journal of Neuroscience Methods* 20:7-21, 1987.
4. **Vanker AD, Stoecker WV:** AI/DERM: Diagnosis of skin tumors. *Proceedings AAMSI Congress 84*, 1984, pp 213-217.
5. **Wasserman GS:** *Color Vision: An Historical Introduction*, New York: John Wiley and Sons, 1978.
6. **Wyszecki G, Stiles WS:** *Color Science: Concepts and Methods, Quantitative Data and Formulae*. New York: John Wiley and Sons, 1982.
7. **Ballard DH, Brown CM:** *Computer Vision*, Englewood Cliffs, NJ: Prentice-Hall, 1982.
8. **Gershon R:** Aspects of perception and computation in color vision. *Computer Vision, Graphics and Image Processing* 32 (2):244-277, 1985.
9. **Horn BKP:** *Robot Vision*, Cambridge: MIT Press, 1986.
10. **Kender JR:** "Saturation, hue, and normalized color: calculation, digitization effects, and use." Department of Computer Science, Technical Report, Carnegie-Mellon University, Pittsburgh, Pennsylvania, 1976.
11. **Robertson TV, Fu KS, Swain PH:** "Multispectral image partitioning." LARS Information Note 071373, Purdue University, West Lafayette, IN, 1973.

12. **Underwood SA, Aggarwal JK:** Interactive computer analysis of aerial color infrared photographs. *Computer Graphics and Image Processing* 6:1–24, 1977.

13. **Horowitz SL, Pavlidis T:** Picture segmentation by a directed split-and-merge procedure. *Proceedings of 2nd International Joint Conference on Pattern Recognition,* Copenhagen, August 1974, pp 424–433.

14. **Ohlander R, Price K, Reddy RD:** Picture segmentation using a recursive region splitting method. *Computer Graphics and Image Processing* 8:313–333, 1978.

15. **Gowda CK:** A feature reduction and unsupervised classification algorithm for multispectral data. *Pattern Recognition* 17 (6):667–676, 1984.

16. **Sarabi A, Aggarwal JK:** Segmentation of chromatic images. *Pattern Recognition* 13 (6):417–427, 1981.

17. **Schacter BJ, Davis LS, Rosenfeld A:** Scene segmentation by cluster detection in color spaces. *SIGART Newsletter,* No. 58, pp 16–17, June 1976.

18. **Hanson AR, Riseman EM:** "Segmentation of natural scenes," in *Computer Vision Systems,* New York: Academic Press, 1978.

19. **Marr D, Hildreth E:** Theory of edge detection. *Proc Roy Soc Lond, B 207,* pp 187–217, 1980.

20. **Stoecker WV, Moss RH:** "Skin cancer recognition by computer vision." Final Project Report for NSF SBIR Phase I Grant #DCR-8460921, 1985.

21. **Tanimoto S, Pavlidis T:** A hierarchical data structure for picture processing. *Computer Graphics and Image Processing* 4:104–119, 1975.

22. **Stein SK:** *Calculus and Analytic Geometry,* New York: McGraw-Hill, 1977, pp 688–693.

23. **Kennedy JB, Neville AM:** *Basic Statistical Methods for Engineers and Scientists,* New York: Harper and Row, 1986.

24. **Umbaugh SE:** *Computer Vision in Medicine: Identification of Variegated Coloring in Digitized Skin Tumor Images.* MS thesis, Electrical Engineering Department, University of Missouri–Rolla, 1987.

25. **Ohta YI, Kanade T, Sakai T:** Color information for region segmentation. *Computer Graphics and Image Processing* 13:222–241, 1980.

Editor's Note: The authors originally submitted their illustrations in color. Due to production limitations, however, we were unable to accept them as such. The authors then kindly submitted black and white versions, and altered their text to conform to this format.

Scott E Umbaugh is currently a Chancellor's Fellow and a graduate research assistant at the University of Missouri–Rolla, where he is completing the Ph.D. degree in electrical engineering. He received the B.S.E. degree with honors from Southern Illinois University at Edwardsville in 1982 and the M.S.E.E. degree from UMR in 1987. Umbaugh worked as a computer software/hardware engineer for ITT North Electric in Columbus, Ohio and McDonnell Douglas Corporation and Affinitec Corporation in St. Louis, Missouri from 1981 through 1986. He is a member of the IEEE Computer Society and the IEEE Engineering in Medicine and Biology Society. His interests include artificial intelligence, computer vision, biomedical applications of computers, and musical applications of microprocessors.

Scott Umbaugh can be reached at the Electrical Engineering Department, University of Missouri–Rolla, Rolla, MO 65401.

Dr. Randy H. Moss is currently Associate Professor of Electrical Engineering at the University of Missouri–Rolla. He received the B.S. (with high honors) and M.S. degrees in electrical engineering from the University of Arkansas in 1975 and 1977, respectively. He was a National Merit Scholar and an NSF Graduate Fellow. He received the Ph.D. degree in electrical engineering in 1981 from the University of Illinois. He joined the faculty of the University of Missouri–Rolla in 1981 as an assistant professor and was promoted to his current rank in 1986. Dr. Moss' research interests are in image processing, pattern recognition, and computer vision. He has developed a machine vision course and laboratory at the University of Missouri–Rolla under a grant from the General Electric Foundation. He is especially interested in vision systems for industrial robots, automated inspection systems, and medical applications of image processing and pattern recognition. He currently serves as an associate editor for both *Pattern Recognition* and *Computerized Medical Imaging and Graphics.* He is a member of IEEE, SME, SAE, and the Pattern Recognition Society and has served as president of the UMR Chapter of Sigma Xi and for two years as Secretary-Treasurer of the Rolla Subsection of the IEEE.

William V. Stoecker practices dermatology in Rolla, Missouri. He received the B.S. in mathematics from Caltech in 1968, the M.S. in systems science from UCLA in 1970, and the M.D. from the University of Missouri–Columbia in 1977. He is Clinical Assistant Professor of Internal Medicine–Dermatology at the University of Missouri–Columbia and Adjunct Assistant Professor of Computer Science at the University of Missouri–Rolla. He is Chairman of the American Academy of Dermatology Task Force on DERM/DDX, which has developed an on-line differential diagnosis system for dermatologists. He is president of Stoecker Moss & Co., which develops medical computer vision systems. His research interests include artificial intelligence in medicine, computer vision in medicine and diagnostic problems in dermatology.

Dr. Stoecker is an associate editor for *Computerized Medical Imaging and Graphics.* He is a member of the UMR Chapter of Sigma Xi.