01 Jan 1996

Unsupervised Color Image Segmentation: with Application to Skin Tumor Borders

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

G. A. Hance

Scott E. Umbaugh

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

Unsupervised Color Image Segmentation

With application to skin tumor borders

Gregory A. Hance[1], Scott E. Umbaugh[1] Randy H. Moss[2] and William V. Stoecker[3]
[1]Department of Electrical Engineering, Southern Illinois University at Edwardsville
[2]Department of Electrical Engineering University of Missouri-Rolla
[3]Department of Internal Medicine, Division of Dermatology, and Department of Computer Science, University of Missouri-Columbia

Detecting boundaries in images is a fundamental problem in computer vision as well as a necessary preliminary step for further image understanding. Our research was developed to serve as a front-end for a computer vision system for skin cancer diagnosis. The most predictive features for various skin cancers will be targeted by the computer vision system, allowing automatic induction software to classify the tumor [8]. The problem of interest in this article is identifying skin tumor boundaries; the border is the first and most critical feature to identify. Object boundaries and surface contours are fairly easily detected by the human observer, but automatic border detection is a difficult problem. The images may contain reflections, shadows or extraneous artifacts that make the process of finding the border more difficult. The images used in this research were digitized from 35-mm color photographic slides obtained from a private dermatology practice and from New York University [8].

Border Finding Algorithm

A skin tumor may be distinguished from surrounding skin by features such as color, brightness or luminance, texture and shape, and any combination thereof. The use of color as a means to identify the tumor border is of particular importance, since in some cases, it is difficult to identify the tumor border in a monochrome image. The border finding algorithm presented here involves a series of preprocessing steps to remove noise from the image, followed by color image segmentation, data reduction, object localization, and contour encoding. This process is depicted in Fig. 1.

Noise Removal

The input image may contain noise that will make the segmentation process less accurate. For example, skin tumor images often contain extraneous artifacts such as rulers and hair that make it more difficult to localize the tumor border (see Fig. 4). In addition, the images may contain undesirable color variations such as shadows and reflections that tend to bias the color map when performing color segmentation. In order to reduce the effects of such noise, the images are first processed with a pseudomedian filter (Fig. 5) and then a nonskin detecting algorithm (Fig. 2) is applied to mask out the unwanted artifacts. As seen in Fig. 6, the majority of the reflections and ruler artifacts have been masked out be this algorithm; the masked out pixels are set to black.

Pseudomedian Filter

Median filtering is computationally intensive, and thus a simpler operator, called the pseudomedian filter [1], which possesses many of the same desirable properties of the median filter, was used. The median of an L-element sequence can be expressed as the MAX (or MIN) of the MIN (or MAX) of all L!/[M!][(L - M)!] subsequences, where M = (L+1)/2 and L! = L(L-1)(L-2)...(2).

The pseudomedian of length five is defined as:

PMED(a,b,c,d,e)=
=(1/2)MAX[MIN(a,b,c),MIN(b,c,d),
MIN(c,d,e)+
(1/2MIN[MAX(a,b,c),MAX(b,c,d),
MAX(c,d,e)]

The MIN followed by MAX contributions of the first part of the equation always result in the actual median or a value smaller, while the MAX followed by the MIN contributions result in the actual median or a value larger. The average of the two contributions tends to cancel out the biases. The pseudomedian definition can be generalized as follows:

$$PMED\{S_L\} = (1/2)MAXIMIN\{S_L\} + (1/2)MINIMAX\{S_L\}$$

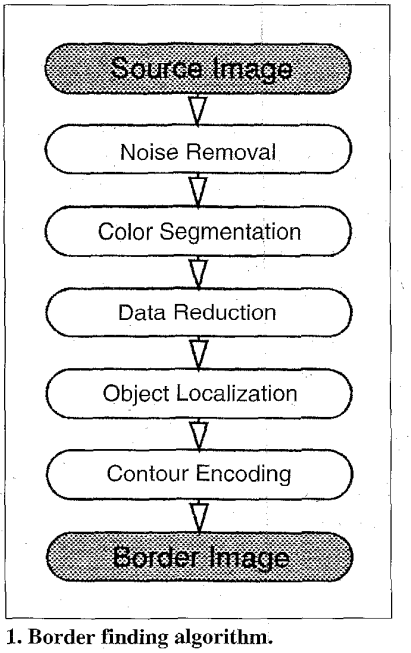
where $\{S_L\}$ denotes a sequence of elements $s_1,s_2,...,s_L$

Nonskin Masking

An algorithm for masking out nonskin is used after the median filter to reduce the effects of having artifacts such as rulers and flash points (bright spots due to reflections) in the image [2]. The algorithm is based on a set of heuristics to determine whether a color is likely to be skin or nonskin. The algorithm compares the brightness level of pixel samples in the green and blue planes to that of the red, and uses predetermined thresholds to decide if the pixel is likely to be nonskin (Fig. 2).

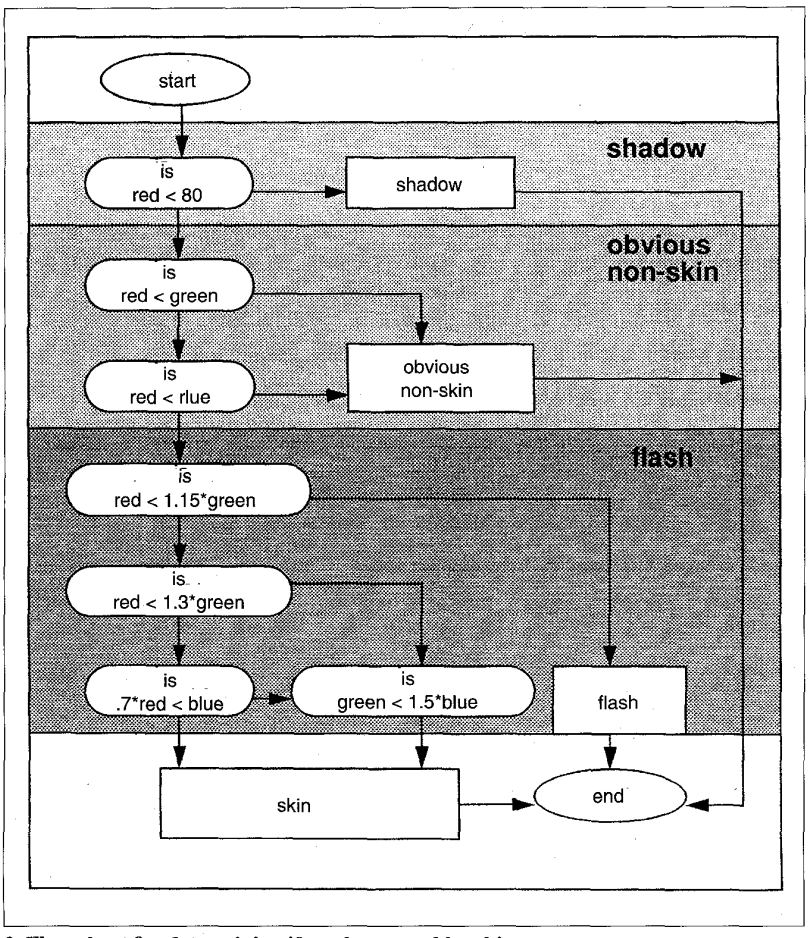
Color Segmentation

The first step in finding the correct border is to segment the image. The border finding algorithm presented here uses color as the basis for segmenting the tumor images into meaningful regions. Six different color segmentation algorithms

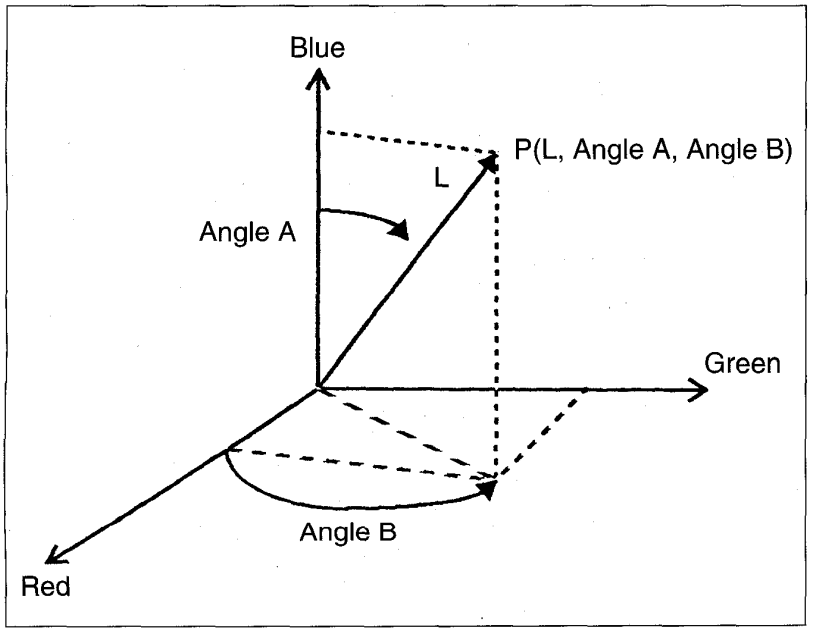


1. Border finding algorithm.

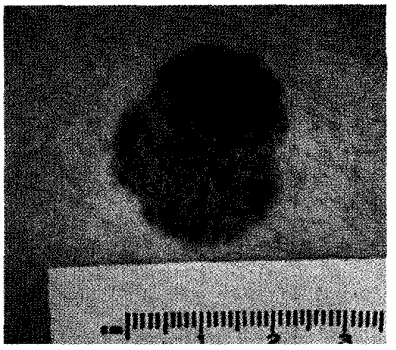
0739-5175/96/$4.00©1996 January/February 1996



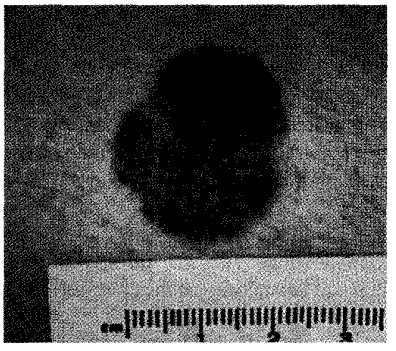
2. Flow chart for determining if a color resembles skin.



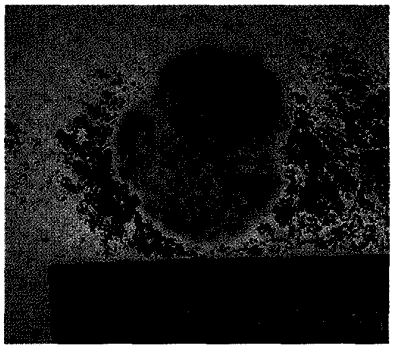
3. The spherical transform.



4. Original tumor image no. 316.



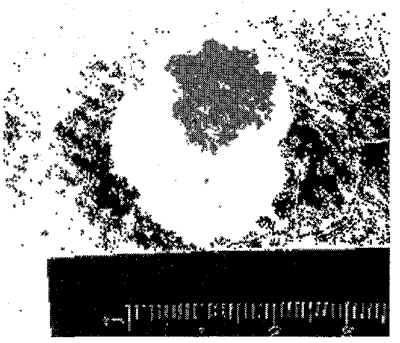
5. Pseudomedian filtered image.



6. Nonskin masked image.

are explored here; adaptive thresholding [3], fuzzy c-means [4], SCT/center split [5], PCT/median cut [5] [6], split and merge [7] and multiresolution segmentation [7].

For all methods, the number of colors for segmentation was kept constant at three, an empirically determined optimum based on error criteria for tumor images [8], with the exception of the spherical transform segmentation method [5] which, by algorithmic definition, segmented the image into four distinct colors. It should be noted that if the image is segmented into too many colors, it may significantly complicate the border find-



7. Adaptive thresholding segmented image.

ing task, while too few colors may result in border information being lost. The idea is thus to find the minimal number of colors while still retaining the maximum amount of border information [5].

Adaptive Thresholding

In performing adaptive thresholding (Fig. 7), the majority of the color information is first mapped into a single image plane using the principal components transform (PCT) [5]. This plane is then used to generate a multiscale description [4] of a histogram by convolving it with a series of Gaussian kernels of gradually increasing width (standard deviation), and marking the location and direction of the sign change of zero-crossings in the second derivative. This process is known as scale-space filtering.

The convolution is given by:

$$F(\xi,\tau) = f(x) * g(\xi,\tau)$$

$$= \int_{-\infty}^{\infty} f(u) \frac{1}{\sqrt{(2\pi)^{\tau}}} \exp\left[\frac{-(\xi-u)^2}{2\tau^2}\right] du$$

where "*" denotes a 1-D convolution.

The parameter (ξ, τ) -space [4] is known to be the scale-space, where τ is the scale constant. The scale-space depends both on the independent variable, ξ, and the Gaussian deviation or scale constant, τ. The scale constant is inversely proportional to the number of peaks and valleys that can be extracted from the histograms. The importance of this fact is that if *a priori* knowledge is available about a particular type of image, τ can be set to be a constant to obtain the desired number of peaks and valleys, and the process can be made unsupervised. The τ value that gives a satisfactory result in most of the color spaces used is 5.

The resulting description, or fingerprint, is interpreted by relating pairs of zero-crossings to modes in the histogram, where each mode is modeled after a normal distribution. After histogram analysis, valid classes are determined from the modes of the histogram and mapped accordingly. Each class represents a different object within the image.

Fuzzy c-Means

Color image segmentation based on the thresholding and fuzzy c-means (see Fig. 8) can basically be divided into two stages: coarse and fine segmentation [4]. The coarse segmentation is intended to reduce the computational burden required for the fine segmentation, i.e., the fuzzy c-means.

In coarse segmentation, a scale-space filter, as described above, is used to analyze the histograms of the three RGB color bands. It determines the number of valid classes and assigns classified pixels to these classes according to the threshold value (safety margins).

The fine segmentation uses fuzzy c-means to assign the remaining unclassified pixels to the closest class. After histogram analysis, valid classes are determined according to the safety margin (usually 5% - 20%) specified by the user. Classified pixels will be assigned to a valid class, and the other pixels will be tagged as unclassified. The larger the safety margin, the more accurate the result of the segmentation will be, and the more computational effort will be needed when using the fuzzy c-means fine segmentation stage. In the fine segmentation stage, the fuzzy membership [4] of the unclassified pixels are calculated and assigned to the class with maximum membership value.

Spherical Coordinates Transform/Center Split

This algorithm, SCT/center 2-D split [10], was initially developed for the identification of variegated coloring [10]. This algorithm consists of transforming the original RGB data into the spherical transform domain, which maps the image into a color space represented by two angles, A and B below, and a one-dimensional intensity (brightness) space L (Fig. 3). The equations to convert from RGB (rectangular) coordinates to spherical coordinates are as follows:

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

$$AngleA = \text{a}\cos\left[\frac{B}{L}\right]$$

$$AngleB = \text{a}\cos\left[\frac{R}{L \times \sin(AngleA)}\right]$$

The two-dimensional color space is then divided using a center split [5], which simply finds the mid-point between the minimum and maximum along each dimension. In this case, the image is split into 4 colors to be able to distinguish the tumor border (Fig. 9).

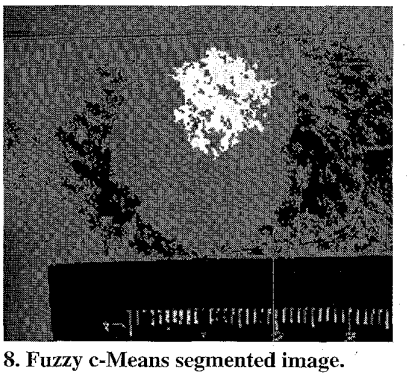
Principal Components Transform/Median Cut

This algorithm is also based on the principal components transform (PCT). In this transform, the eigenvectors of the covariance matrix are used as a linear transform matrix on the original [RGB] vectors, so that the resulting vectors have components that are uncorrelated. Geometrically, this means that the primary axis has been aligned where the variance in the data is maximal. The new vectors, here called $[X_1\ X_2\ X_3]^T$, are obtained by the following equation:

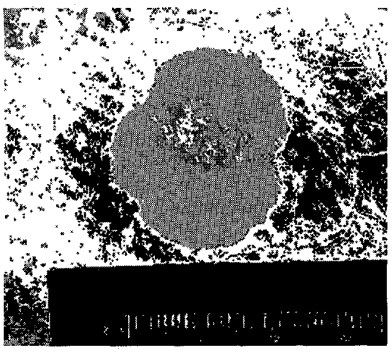
$$\begin{bmatrix} X_1 \\ X_2 \\ X_3 \end{bmatrix} = \begin{bmatrix} E_{11} & E_{12} & E_{13} \\ E_{21} & E_{22} & E_{23} \\ E_{31} & E_{32} & E_{33} \end{bmatrix} \bullet \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

where $[E_{11}\ E_{12}\ E_{13}]$, $[E_{21}\ E_{22}\ E_{23}]$ and $[E_{31}\ E_{32}\ E_{33}]$ are the eigenvectors of the three-dimensional (RGB) color covariance matrix of the image [5].

The median cut [6] component of the segmentation method is based on an algorithm that was developed for color compression; specifically, to map 24-bit-per-pixel color images into images requiring an average of 2 bits per pixel. The motivation behind this algorithm is to be able to display high-quality reproduc-



8. Fuzzy c-Means segmented image.



9. SCT/Center Split segmented image.

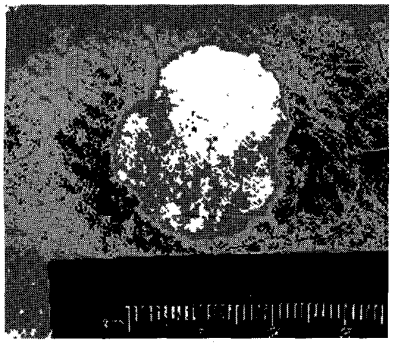
tions of color images with small frame buffers. In this experiment, supported by previous research results [8], it was determined that segmentation into three colors was generally enough to allow for automatic determination of the tumor border (see Fig. 10).

The color quantization task includes four phases:

1. Sampling the original image for color statistics
2. Choosing a colormap based on the color statistics
3. Mapping original colors to their nearest neighbors in the colormap
4. Quantizing and redrawing the original image

Split and Merge Segmentation

Split and merge segmentation techniques are based upon a quad tree data representation, whereby a square image segment is broken (*split*) into four quadrants if the original image segment is nonuniform in attribute, as determined by a predefined homogeneity criteria based predicate test. If four neighboring squares are found to be uniform, then they are replaced (*merge*) by a single square composed of the four regions. The predicate test used to determine region uniformity



10. PCT/Median Cut segmented image.

was defined as follows—a region is considered to be uniform if the local variance of the region is less than or equal to the global variance of the image. Thus, a region is considered to be nonuniform and subject to splitting if the local variance is greater than the global. Individual pixels not satisfying the criteria are assigned a zero value.

In principle, the split and merge process could start at the full image level, and initiate split operations. This approach tends to be computationally intensive when the predicate test (homogeneity criterion) involved cannot be easily satisfied with a gross information-first approach. Conversely, beginning at the individual pixel and making initial merges has the drawback that region homogeneity measures are limited at the single pixel level. Initializing the split and merge process at an intermediate level enables the use of more powerful homogeneity tests, without excessive computation.

The algorithm calls for the basic traversal procedure of a $2^n \times 2^n$ image starting at level q, which contains 2^{2q} squares of size $2^{n-q} \times 2^{n-q}$ each. (Thus, the entire image corresponds to level 0, and the single pixel level is n.) The algorithm does not create the whole tree, but only the needed parts. It maintains a list, L, whose entries correspond to nodes of the quad tree, initially arranged in such a way that four successive nodes have a common parent in the tree.

Before performing the split and merge segmentation, the color information is first mapped onto a single image plane using the PCT. Once mapped, the average brightness level is determined for the plane and used in the predicate test of the split and merge procedure. The predicate test utilized here determines the standard deviation of each region to determine uniformity, and compares uniform regions with the average brightness level to determine to which class the region belongs, the background or the object. The algorithm continues to split regions until all regions satisfy the homogeneity criterion (Fig. 11).

Multiresolution Segmentation

The multiresolution segmentation [7] method is similar to the split and merge segmentation, except that there is no merging, and splitting begins at the top level. The same predicate test used for the split and merge algorithm is used here as well (Fig. 12).

Data Reduction

Once the image has been segmented, it contains many objects of varying sizes represented by a small range of colors. Many of the objects are not of interest and can be eliminated from the image. The objects are also often connected together by narrow isthmuses and contain jagged protrusions. The next step is to label and be able to distinguish among all of the objects present in the image. This is a computationally intensive process, and it is advantageous to reduce the number of objects being labelled as well as to separate them from each other. At this point, the morphological attributes of objects within the image are of the importance, and for simplicity the segmented color images can be represented in terms of grayscale, where each color is mapped to a different grayscale pixel value.

A grayscale morphological opening procedure [8] was performed first to smooth the contours of the object, break narrow isthmuses, and eliminate thin protrusions and small objects. The opening of set A by a structuring element [9] B, as defined below, says that the opening of A by B is the erosion of A by B, followed by a dilation of the result by B, where the dilation of an image simply expands it and erosion shrinks it.

$$A \odot B = (A \ominus B) \oplus B$$

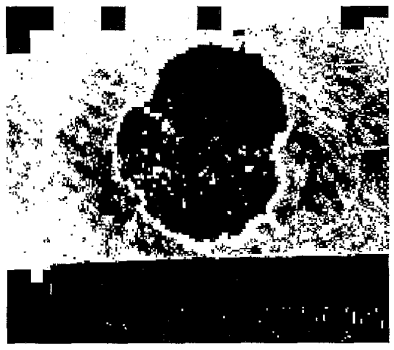
Following the opening procedure, grayscale morphological closing was performed to fill gaps in the contour and eliminate small holes. The closing of set A by structuring element B, defined below, says that the closing of A by B is the dilation of A by B, followed by the erosion of the result by B.

$$A \bullet B = (A \oplus B) \ominus B$$

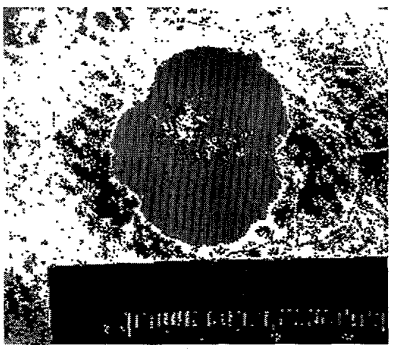
The result of applying this morphological opening-closing procedure to Fig. 13 is shown in Fig. 14.

Object Localization

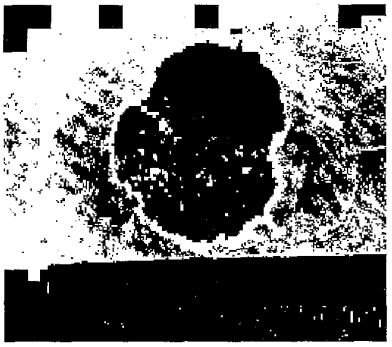
Once the object data has been reduced and smoothed, it is necessary to label all of the objects present in the image and determine which object is most likely to be the tumor object. A sequential labeling algorithm is used to label the objects. If no objects are found, the image is represented empirically as an empty, set and all pixel values are set to 1, which represents an image containing background only and no tumor. Once the objects are labeled then the zero, first, and second order moments



11. Split and Merge segmented image.



13. SCT/Center Split segmented image with noise.



12. Multiresolution segmented image.

14. Segmented image after morphological data reduction.

[1] of each object are calculated. Next, the moments are used to determine the area, centroid and eigenvalue ratio of each object, which are used to trim the list of objects and leave only those that are possible tumor candidates. This selection s based on the assumption that the tumor should exhibit some degree of circularity [1] and also be of significant size relative to the image. The ratio:

$$R_A = \frac{\lambda_N}{\lambda_M}$$

(λ_N = eigenvalue for minor axis, λ_M = eigenvalue for major axis) of the minor-to-major axes for the moments of inertia comprise the eigenvalue ratio of an object, which is a useful shape feature as a measure of circularity. It was empirically determined that objects smaller than 0.5% and larger than 90% of the image size could be discarded, and objects with an eigenvalue ratio of less than 0.1 could also be eliminated from consideration. From this list of objects, a border candidate is selected which has its centroid closest to that of a manually determined point that is approximately the centroid of the original tumor. In the final application this step will not be necessary, as the tumor will be in the center of the image.

The process of object localization consists of four phases:

1. object labeling
2. calculating object properties (area and circularity)
3. trimming the object list
4. choosing the best tumor object candidate

Contour Encoding

Once the tumor object candidate has been selected, it is necessary to encode the object contour. This is accomplished by using Freeman chain coding [1] to follow and vectorize the contour. Once the border contour has been vectorized (Fig. 15), the contour is smoothed by subsampling the vector data and using a B-spline to connect the points (Fig. 16). A seed fill algorithm is then used to fill in the border contour (Figs. 17 and 18).

Experimental Results

The following error metric was used to determine the success of the border segmentation relative to the true border. The true border for each tumor was determined manually by a dermatologist.

Let A_{ij} be the data set representing the actual (manual) border.

Let B_{ij} be the data set representing the segmented border.

A_{ij} and B_{ij} are both binary images, 0 = tumor, 1 = background:

$$\text{Border Error Metric} = \frac{area(A_{ij} \otimes B_{ij})}{area(A_{ij})},$$

where $\otimes$ is the exclusive-or operation

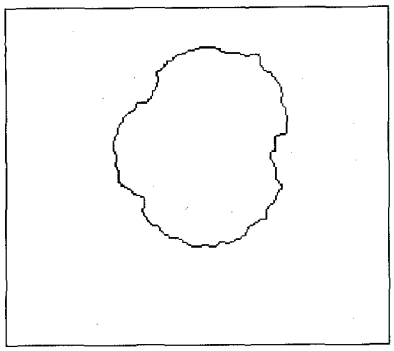
Using this metric, a value of 0 will result if both the manual and segmented border are exactly the same, and a value of 1 will results if the segmented border is the empty set, as would occur with an algorithm that found no information. Thus, the useful range of values will be in the range (0.0 to 1.0). The Table presents the average error and standard deviation of the border finding process with respect to each color segmentation algorithm.

Figure 19 illustrates a comparison of the result of the border finding algorithm for each of the six color segmentation methods. Each image in Fig. 19 depicts the original skin tumor image number 365, with an overlay of the detected border for the respective segmentation method implemented. The PCT/median cut segmentation method shown in Fig. 19d resulted in the most accurate border for this tumor, with an error metric of 0.105. The constraint of an upper limit of 1.0 has been incorporated into tabulating the data. The results show that the best color segmentation algorithm used in conjunction with the border finding process is the PCT/median cut. In addition to the employment of each individual segmentation algorithm as part of the border finding process, a method for combining the segmentation algorithms was explored. This combined method involves simply merging information from each of the six segmentation methods at the object localization stage of the border finding algorithm. The border objects resulting from each of the segmentations are compared, and the object that most accurately conforms to the criteria for an ideal border candidate is chosen. Although this combined segmentation is more computationally expensive, it is trivial to implement once the individual segmentation algorithms have been developed. It inherently increases the likelihood of correctly identifying the tumor border.

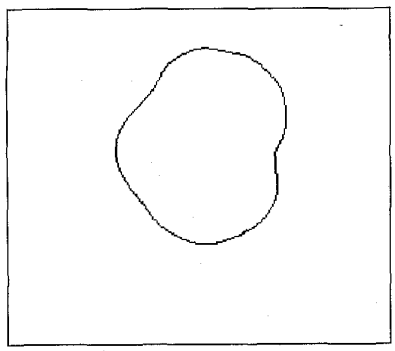
The error distributions are illustrated in Fig. 20.

Conclusion

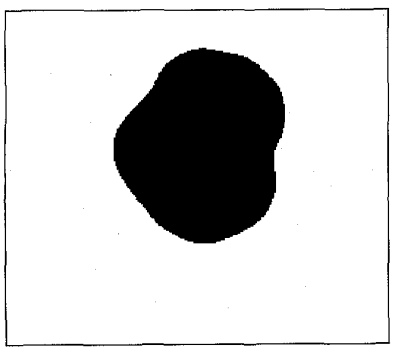
This article compared six different



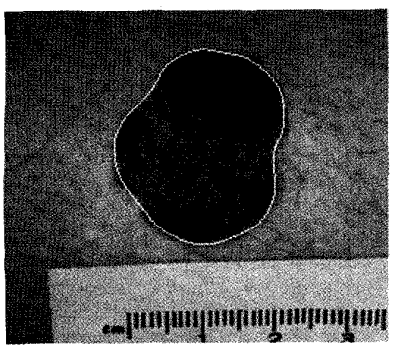
15. Freeman chain encoded border contour.



16. Splined border contour.



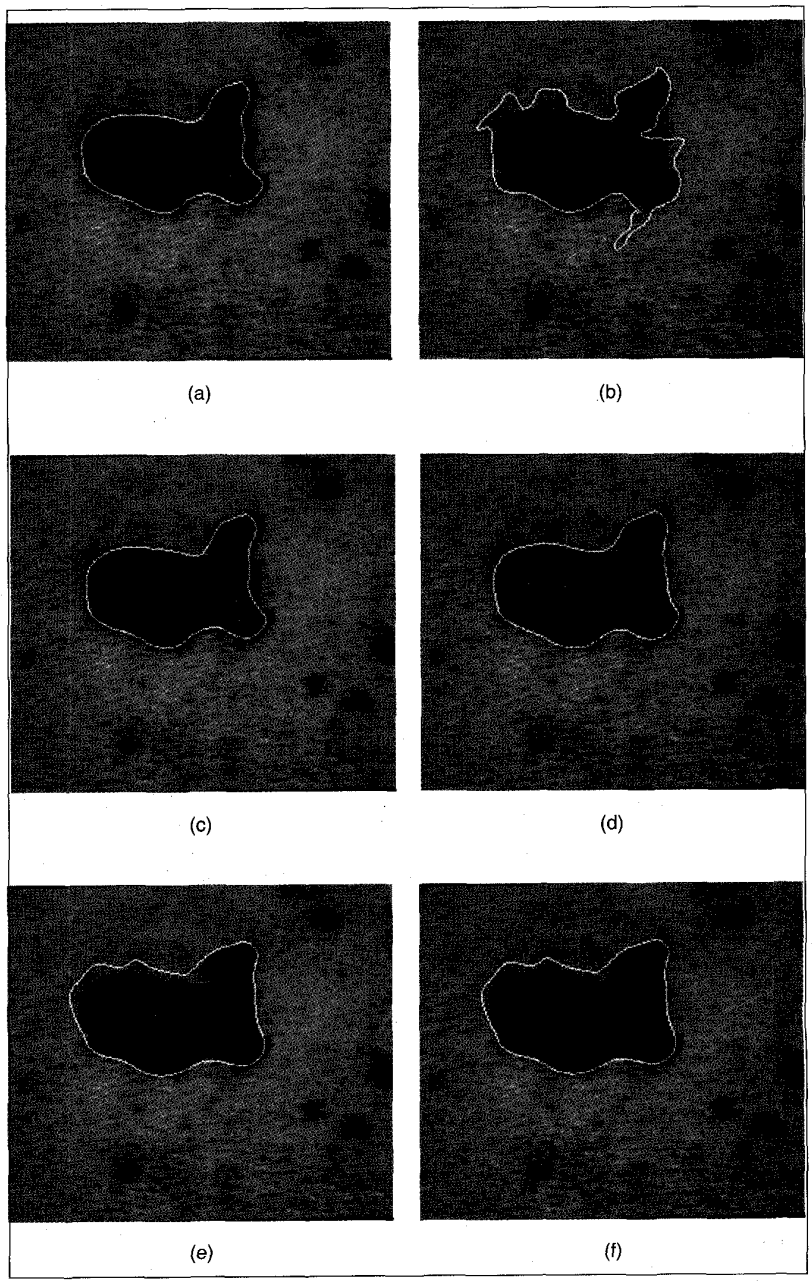
17. Filled contour.



18. Original image with splined border.

color segmentation methods and their effectiveness as part of an overall border finding algorithm. The PCT/ median cut and adaptive thresholding algorithms provided the lowest average error and show the most promise for further individual algorithm development. Combining the different methods resulted in further improvement in the number of correctly identified tumor borders, and by incorporating additional heuristics in merging the segmented object information, one could potentially further increase the success rate.

The algorithm is broad-based and suggests several areas for further research. One possible area of exploration is to incorporate an intelligent decision making process as to the number of colors that should be used for segmentation [5] in the PCT/ median cut and adaptive thresholding algo-



19. Comparison of resulting border contours using six different segmentation methods, error measure in parentheses: (a) adaptive thresholding (0.113), (b) SCT/Center Split (0.292), (c) Fuzzy c-Means 0.109), (d) PCT/Median Cut (0.105), (e) Multiresolution segmentation (0.238), (f) Split and Merge (0.238).

Table: Error Metric Average and Standard Deviation			
Algorithm	Average Error	Standard Deviation	Number of tumors with error < 1 (66 total)
ATS (adaptive thresholding)	0.665	0.336	40
CSS (SCT/center split)	0.797	0.322	25
FCS (fuzzy c-means)	0.702	0.330	42
MCS (PCT/median cut)	0.524	0.372	46
MRS (multiresolution)	0.770	0.343	23
SMS (split and merge)	0.782	0.338	22
Combined Method	0.367	0.263	57

rithms. For comparison purposes, the number of colors was kept constant at three in our application. Other areas that can be explored are noise removal and object classification to determine the correct tumor object.

Acknowledgment

This work was supported in part by NIH grant #R43 CA60294-0 and SIUE Funded University Research Award #F-EN310.

We thank Mark Zuke for his assistance in preparing the figures.

References

1. **Pratt, William K:** *Digital Image Processing: Second Edition* New York: John Wiley and Sons, 1991.

2. **McLean, Robert:** *Tumor Classification based on Relative Color Analysis of Melanoma and Non-Melanoma Tumor Images*, M.S. Thesis, Department of Electrical Engineering, University of Missouri-Rolla, 1994.

3. **Carlotto MJ:** Histogram Analysis Using a Scale-Space Approach. *PAMI*, 1(9): 121-129, 1987

4. **Lim YW, Lee SU:** On the Color Image Segmentation Algorithm Based on The Thresholding and The Fuzzy c-Means Techniques. *Pattern Recognition*, 23(9):935-952, 1990.

5. **Umbaugh, Scott E, Moss, Randy H, Stoecker, William V, Hance, Gregory A:** *Automatic Color Segmentation Algorithms: With Application to Skin Tumor Feature Identification*, IEEE Engineering in Medicine and Biology, 12(3): 75-78, 1993.

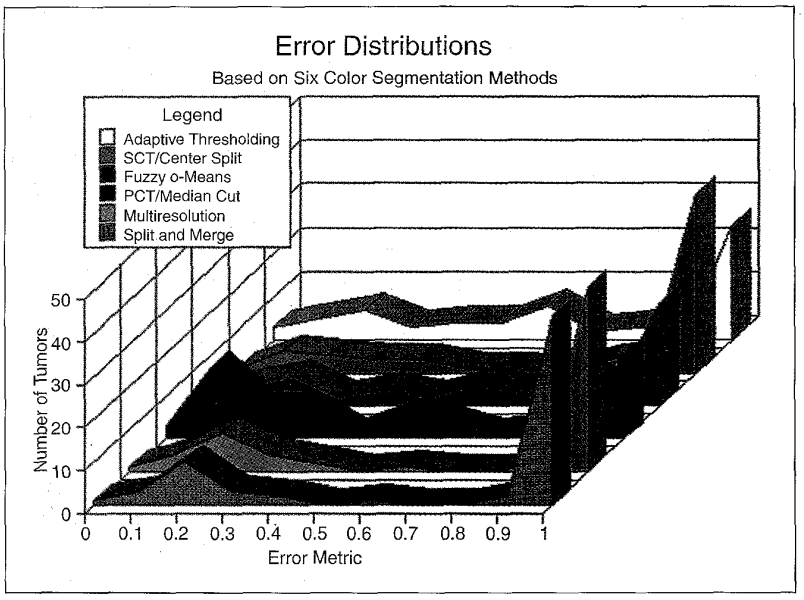
6. **Heckbert P:** Color Image Quantization for Frame Buffer Display. Computer Graphics, 16(3):297-304, 1983.

7. **Pavlidis, Theodore:** *Algorithms for Graphics and Image Processing*, Computer Science Press, Roclsville, MD, 1982.

8. **Umbaugh, Scott E, Moss, Randy H, Stoecker, William V:** *An Automatic Color Segmentation Algorithm with Application to Identification of Skin Tumor Borders*, Computerized Medical Imaging and Graphics, 16(3): 227-235, 1992.

9. **Gonzales RC, Woods RE:** *Digital Image Processing*. Canada: Addison Wesley Publishing Co., 1992.

10. **Umbaugh, Scott E, Moss, Randy H, Stoecker, William V:** *Automatic Color Segmentation of Images with Application to Detection of Variegated Coloring in Skin Tumors*, IEEE Engineering in Medicine and Biology, 8(4): 43-52, 1989.



20. Error distributions of the six segmentation methods. The distribution for the PCT/median cut segmentation algorithm, which proved to be best for skin tumor border detection, has a large number of tumors in the lower error metric range (0-0.4) and a smaller amount with an error metric of 0.9 or greater. The SCT/center split, split and merge, and multiresolution algorithms have a larger number of tumors with error metric 0.9 or greater and with few tumors in the lower error metric range. The adaptive thresholding and fuzzy c-means algorithms result in error distributions that are more evenly distributed with a relatively small cluster of tumors in the 0.9 to 1.0 range. These distributions indicate that while the PCT/median cut algorithm shows the most promising results, the algorithms with more evenly distributed errors, adaptive thresholding, and fuzzy c-means, also have potential for future research efforts.

Scott E. Umbaugh is currently an Associate Professor in Electrical Engineering at Southern Illinois University at Edwardsville. He received the BSE degree with honors from SIUE in 1982, the MSEE from the University of Missouri-Rolla in 1987, and the PhD in electrical engineering from UMR in 1990. While at UMR he was a Chancellor's Fellow. He worked in industry as a computer design engineer from 1981 through 1986, and as a computer vision and image processing consultant from 1986 through 1995. He currently serves on the Editorial Board for the *IEEE Engineering on Medicine and Biology Magazine* and the *Pattern Recognition* journal. He is a member of the IEEE Education Society, the IEEE Engineering in Medicine and Biology Society, the Pattern Recognition Society, the American Society for Engineering Education, Eta Kappa Nu, and Sigma Xi. His professional interests include artificial intelligence, com-

puter vision, medical applications of image processing, and electrical engineering design education. Dr. Umbaugh can be reached at the Department of Electrical Engineering, Southern Illinois University at Edwardsville, Edwardsville, IL, 62026-1801, or seu@ee.siue.edu.

Gregory A. Hance is currently a computer software/systems engineer for SESI (Systems Engineering Solutions, Inc.) involved in Work Flow application development and UNIX systems programming. He received the BSE degree with honors from SIUE in 1991, where he served as a member of IEEE and president of the electrical engineering honor society (Eta Kappa Nu). He is currently completing his thesis involving the development of a comprehensive computer vision/image processing software tool box to statisfy the requirements to receive his MSEE degree from SIUE. He is a member of the the IEEE Engineering in Medicine and Biology Society and Computer Society. His professional interests include artificial intelligence, computer vision, medical applications of image processing, network engineering and work flow automation. Greg Hance can be reached at the Department of Electrical Engineering, Southern Illinois University at Edwardsville, Edwardsville, IL, 62026-1801, ghance@ee.siue.edu.

Randy H. Moss is Professor of Electrical Engineering at the University of Missouri-Rolla(UMR). His research interests are in the areas of image processing, pattern recognition, and computer vision. He is especially interested in medical and industrial applications of machine vision techniques. Dr. Moss received the B.S.E.E. and M.S.E.E. degrees in Electrical Engineering from the University of Arkansas where he was a National Merit Scholar and the Ph.D. from the University of Illinois, where he was a National Science Foundation Graduate Fellow. He serves as an Associate Editor of Pattern Recognition and of Computenzed Medical hnaging and Graphics. His awards include the 1988 Society of Automotive Engineers Ralph R Teetor Educational Award, the Society of Manufacturing Engineers 1987 Outstanding Young Manufacturing Engineer Award, honorable mention for the 1986 Eta Kappa Nu C. Holmes MacDonald Outstanding Teaching Award, the Tenth Annual Pattern Recognition Society Award (awarded in 1985 for the best paper published in Patted Recognition in 1983), eight UMR Faculty Excellence Awards, and a UMR Outstanding Teacher Award. Dr. Moss is a senior member of I.E.E.E. and a member of Sigma Xi, the Pattern Recognition Society, Eta Kappa Nu, Tau Beta pi, and Phi Kappa Phi.

William V. (Van) Stoecker received the B.S. degree in Mathematics in 1968 from the California Institute of Technology, the M.S. in Systems Science in 1971 from the University of California, Los Angeles, and the M.D. in 1977 from the University of Missouri, Columbia. He is Adjunct Assistant Professor of Computer Science at the University of Missouri-Rolla and Clinical Assistant Professor of Internal Medicine-Dermatology at the University of Missouri-Columbia. His interests include computer-aided diagnosis and application of computer vision in dermatology.