

ROBUST VOLUME CALCULATIONS OF TUMORS OF VARIOUS SIZES¹

Adele P. Peskin,^{(a)(d)} Karen Kafadar,^(b) A.M. Santos,^(c) and Gillian G. Haemer^(a)

^(a) Scientific Application and Visualization Group

National Institute of Standards and Technology, Boulder, Colorado 80305 USA

^(b) Department of Statistics and Physics, Indiana University, Bloomington, Indiana 47408 USA

^(c) Department of Mathematical and Statistical Sciences, University of Colorado–Denver 80217 USA

^(d) Corresponding author: peskin@boulder.nist.gov

Abstract - *We present a statistical method that leads to accurate volume measurements of lung tumors from computerized tomographic (CT) data. The method is based on the assumption that a range of pixel intensities in CT data defines the edge of a tumor, and from our statistical model, we assign a probability that a given pixel intensity is included in the tumor measurement. Using the magnitude of the gradient of the pixel intensities over the density range observed for lung tumors and lung tissue, we have found consistent metrics that help define these weights, so that the measurement does not require user-controlled parameters and can be performed automatically. This could ultimately lead to direct comparisons of measurements from different medical laboratories.*

Keywords: image processing, segmentation, biweight, robust methods

1 Introduction

Tumors in the lung are classified according to their detected growth in CT scans taken over a period of time. Tumors that are large enough to be detected in CT scans vary in size, but often are so small that a large proportion of the tumor pixels lie near the tumor surface, adding to the difficulty of making volume measurements [1, 2]. If an edge of a tumor lies between two pixel locations, radiologists must determine which of those pixels should be included in a measurement of the size of the tumor, determinations which can have large effects on estimated tumor volumes. Current techniques to measure these “partial volumes,” or 3-D voxels in the grid that are only partially filled, in this case by a tumor in a scan of the lung, vary widely in resulting tumor volume measurements. Current techniques to segment objects within an image fall into several different categories, and vary in their approaches to determining partial volumes. Many of these techniques are histogram-based and hence depend on the overall distribution of pixel values in the image. A single threshold or set of threshold values is chosen based on the distribution of pixel

intensities or the distribution of the magnitude of the gradient of the pixel intensities [3]. Techniques also vary according to whether edges are defined by choosing cutoff values, by growing regions from seed points based on cutoff values, or by growing regions with model-based approaches. The goal for each approach is the same: accurate estimates of tumor volumes leading to accurate estimates of tumor growth over time. Most current techniques use associated parameters as a basis for calculations that are input by the user and hence results vary from user to user [4, 5]. We present a technique that involves no user defined parameters.

Our approach is based on the observed distribution of pixel intensities in a region containing the suspected tumor. Tumor edges appear as 3D voxels with partial volumes in CT scans, so the contribution to the volume calculation for these voxels should carry less than the full weight of a filled voxel. A challenge arises because the pixel intensities at tumor edges can be distributed rather widely. We use a statistical approach to weight each pixel intensity according to its “likelihood” of being inside or outside of the tumor. Instead of specifying an exact mask that represents the tumor, we sum the contributions of weighted partial volumes. Each pixel intensity lies in one of three different pixel intensity distributions for the lung pixels: the tumor pixels, the edge pixels (which vary over the entire range), and the background pixels. We describe the data that motivated this work, our method for assigning pixel weights, and then use them to estimate tumor volumes.

2 Data Description

Images were obtained from the Public Lung Database to Address Drug Response, a project which is funded by the Cancer Research and Prevention Foundation (www.via.cornell.edu/crpf.html). We have examined many sets of lung tumor data, and have compared the pixel distributions in the regions of lung tumors using both statistical and visualization tools. We have looked, in particular, at the distribution of pixel intensities within a two-pixel length distance of the tumor edge, where all of the

¹ ¹This contribution of NIST, an agency of the U.S. government, is not subject to copyright.

partially filled voxels occur. It is clear that edges of the tumors lie at varying distances to their closest pixel location, since the grids of pixel locations are laid arbitrarily across the lung data. If the average pixel intensity of an object is x and the average background value is y , a pixel at the edge of that object will be assigned an intensity somewhere between x and y depending upon its relationship to the grid. A method for estimating tumor volume should give consistent results regardless of grid shifts with respect to the tumor. Thus, a single threshold value cannot define the edges of an object.

To visualize the range of pixel values for the edge pixels of the lung tumors, we examine a set of data containing a lung tumor, a slice of which is shown in Figure 1a. We isolate a region of lung CT data around the tumor and examine the distribution of pixel intensities corresponding to tumor interior, tumor edge, and background. Figure 1b displays individual pixel intensities in the region of the tumor, i.e. a discretized version of the data shown in Figure 1a. The edge pixels, color-coded according to pixel intensity, are shown as orange, red, and yellow boxes in this picture, with intensities that range from -650 to -150 Hounsfield units (HU), compared with the interior of the tumor, colored white and grey, in the -150 to 150 HU range, and the background, colored green, blue, and purple, in the -1024 to -650 HU range. A blood vessel that is attached to the tumor on the left is also visible in this figure, as well as cross-sections of other blood vessels.

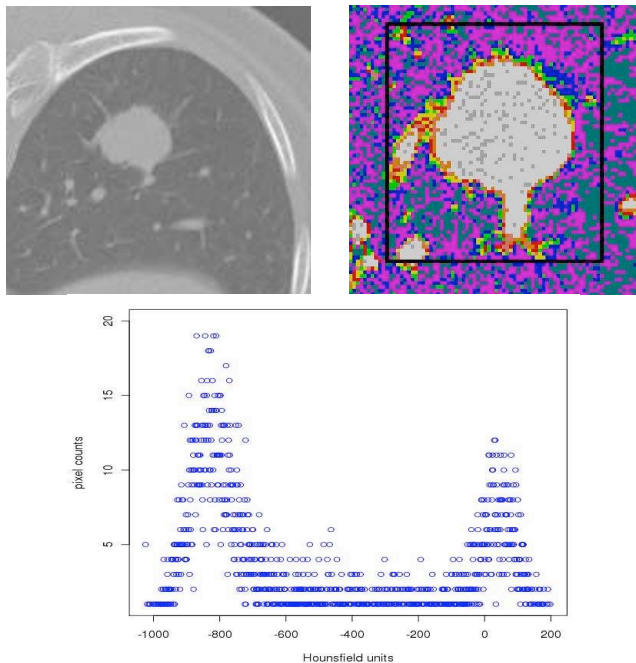


Figure 1. a.) Section of a slice of lung CT data containing a lung tumor; b.) Pixel intensities in Hounsfield units for data in 1a: white:-150 to 100,orange:-250 to -150,pink:-350 to -250,red:-450 to -350,yellow:-550 to -450,green:-650 to -550,blue:-750 to -650,purple:-850 to -750,teal:less than -850; c.) Histogram of the intensities inside the box in 1b.

Figure 1c shows a histogram of pixel intensities in the bounding box of Figure 1b, which clearly indicates two distinct pixel distributions, one for the tumor (centered around 50 HU) and one for its surroundings (centered around -800 HU). Pixel intensities between the modes of these distributions could be either tumor pixels or background pixels. We calculate a weight that each individual intensity comes from one or the other of these distributions, based on the assumption that each pixel has some probability of being tumor or background.

3 Approach

The overall goal of this project is to estimate tumor volumes accurately from CT measurements. Since our data consists of pixel intensities and pixel locations, we need a good understanding of how pixel intensities around a tumor are distributed. We assume the existence of an algorithm to identify the approximate tumor region. We model the distributions using a 3-component Gaussian mixture via the EM (Expectation-Maximization) algorithm. A 2-component mixture model, with a tumor pixel component and a background pixel component, did not accurately represent the data. Thus we reasoned that the edge pixels, whose intensities cover a wide range of pixels, form their own distribution, and expanded the model to include a third component whose standard deviation is much larger than that of the other two components.

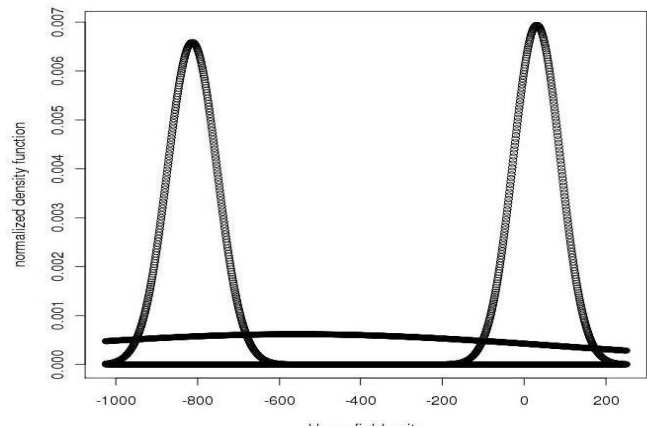


Figure 2. Normalized density functions for the 3 components of pixels from Fig. 1.

The 3-component model illustrated in Figure 2 reliably reflects the actual data. The equations for fitting the 3-component model are described in general terms in [6] and in more detail in Appendix 1. For the normalized density functions shown here, the tumor distribution has a mean value of 26.65 HU and a standard deviation of 51.46 HU. The background distribution has a mean value of -837.43 HU and a standard deviation of 59.27 HU. The volume fractions for the tumor, background, and edge distributions are respectively 0.082312, 0.410151, and 0.507537.

For a given tumor we find the mean and standard deviation of each component of a 3-component mixture. The challenge is

then to determine which pixels from the edge pixel distribution are part of the tumor, and which are part of the background; the former (tumor pixels) should receive full weight (1.0) while background pixels should receive no weight (0.0). Edge pixels receive partial weights in a smooth way, depending upon their distances from the tumor pixel distribution. A smooth weight function that achieves this purpose and has demonstrated excellent performance in many applications is the biweight function [7,8,9]:

$$w(u) = (1 - u^2)^2 I_{[-1,1]}(u), \quad u = (x - T) / (kS) \quad (1)$$

where x is a value for a pixel intensity, T is an estimate of the average of the distribution of tumor pixel intensities, S is an estimate of the scale (e.g., standard deviation) of this distribution, $I_{[-1,1]}$ is the indicator function, and k is a “tuning constant” (typically chosen to be in the range 4–6). Values of x that are more than kS away from the presumed center of the distribution (T) receive zero weight. For our situation, pixel intensities above a certain value are always presumed to be those from tumor, so we need the weight function for only those observations whose pixel intensities are below some cutoff. For these data, we assume that this cutoff is chosen to be $T - k_1S$; i.e., all values x of the pixel intensities greater than $T - k_1S$ receive a weight of 1.0; the values below $T - k_1S$ receive a weight in accordance with the biweight function, until a value $T - k_2S$, beyond which the weight is 0.0. The constants k_1 and k_2 are functions of the data set (i.e., not user-defined), and are described below. First, we recall that the biweight estimate of location, T , from a set of pixel intensities $\{x_1, \dots, x_n\}$, can be calculated as an iteratively re-weighted mean [10]:

$$T_j = \frac{\sum_{i=1}^n x_i w(u_i)}{\sum_{i=1}^n w(u_i)}, \quad w(u_i) = (1 - u_i^2)^2 I_{[-1,1]}(u_i), \quad (2)$$

$$u_i = (x_i - T_{j-1}) / (cS)$$

(typically $c = 6$), $I_{[-1,1]}(u_i)$ is the indicator function that is 1 for $-1 \leq u_i \leq 1$ and zero otherwise, and the iteration over $j = 1, 2, \dots$, continues until convergence to some criterion (e.g., relative change $(T_j - T_{j-1}) / T_j < 10^{-4}$). The iteration often starts with T_0 and S as the median and the median absolute deviation from the median, respectively. [The algorithm actually uses for S a more refined estimate of the scale, described in [10].]

Rather than using the bisquare weight function to estimate the location T of the population from a sample of values, we use the bisquare weights directly for estimating tumor volumes. Because $w[\cdot]$ traditionally is defined as 1 at 0 and 0 at -1 and 1, we needed to translate and rescale the function, and thus we considered two possible weight functions derived from the bisquare weight function:

$$\begin{aligned} w_1(u) &= 0 & u < -1 \\ &= (1 - u^2)^2 & -1 \leq u \leq 1 \\ &= 1 & u > 1 \\ u &= [x - (T - k_1S)] / (k_2S) \end{aligned}$$

$$\begin{aligned} w_2(u) &= 0 & u < -1 \\ &= (1 - u^2)^2 / D & -1 \leq u \leq 1 \\ &= 1 & u > 1 \\ u &= [x - (T - k_1S)] / (k_2S), \\ D &= (k_1 - T) / (k_2S) \end{aligned}$$

The weight function w_1 translates the weight function so it takes the value 1 at $x = (T - k_1S)$ (instead of at $x = T$) and 0 at $x = (T - k_2S)$. The second weight function w_2 takes that part of the ordinary weight that lies in $x = T - k_1S$ and $x = k_2S$ and rescales the values so that $w_2(u) = 1$ when $x = T - k_1S$. In practice, we found rather little difference in the volume estimates between weights defined by $w_1(\cdot)$ versus $w_2(\cdot)$; the results shown below use $w_2(\cdot)$. A plot of the second weight function is shown in Figure 3.

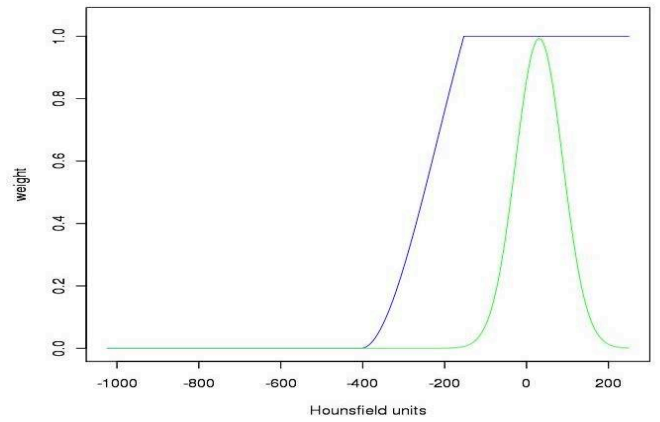


Figure 3. Weighting function (in blue) for edge pixels; tumor distribution (in green).

In our algorithm below, we use for T the estimated mean of the tumor pixel intensity distribution from the EM algorithm. Our algorithm could use instead the biweight estimate from (2), but the biweight assumes as least half of the observations come from the target distribution (in this case, intensities corresponding to tumor). In our data, most intensities are background, so our algorithm would require a user-specified limit below which intensities would be ignored (to ensure a majority from tumor pixels). Since automation is a desirable feature of our algorithm, we chose the first route instead. Below we describe the algorithm for estimating tumor volume using these robust biweight weights.

4 Biweight Constants

We examined 10 independent sets of lung tumor data from two different sources to assess consistency in the definition of the two constants, k_1 and k_2 , needed for the biweight equations above across various data sets. We found significant variation in the pixel intensity distributions that characterize the tumors in these sets, with means ranging from -23.3 to 69.5 HU, and standard deviations ranging from 50.1 to 248.8 HU. However, since our main objective in using these equations is to assign weights to intensities of those pixels that lie at the edges of the

tumors, we reasoned that these two constants, k_1 and k_2 , should be a function of the edge pixel intensity distribution. We expect the edge distributions to be similar across data sets, even though the densities of the tumors may vary, since there is a large, consistent change of approximately 800 HU between the means of these distributions. Therefore, we studied the edge distributions in relation to the change in the gradient field of the pixel intensities at the tumor edges. At each pixel location, we calculated the magnitude of the gradient of the pixel intensity, and for each pixel intensity in the set, we calculated an average value for the gradient magnitude. Examples are shown in Figure 4, in which for 3 of the 10 representative data sets, we display a histogram of the pixel intensities overlaid with a plot of average gradient magnitude values. We used the biweight estimator [10] to estimate a mean and standard deviation for each of the gradient magnitude curves, using values associated with pixel intensities between -800 and -100 HU. These biweight means, as opposed to the EM-estimated means and standard deviations for the tumor pixel distributions, were more similar; see Table 1. The results suggest that the constants k_1 and k_2 in weight function $w_2(\cdot)$ can come directly from the distribution of gradient magnitude values.

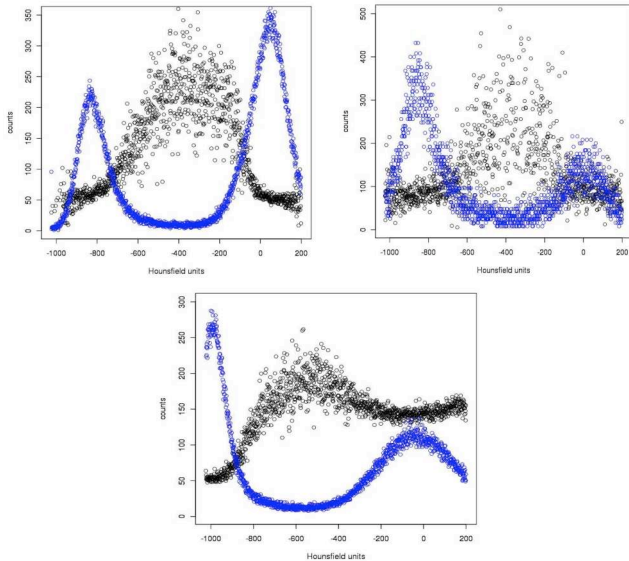


Figure 4. Histograms of pixel intensities in the region of 3 lung tumors (in blue) shown with plots of average gradient magnitudes of the pixel intensities over the same regions (in black) for three representative data sets.

Peskin et al. [11] describe an algorithm for generating realistic synthetic tumor data sets, with features similar to clinical tumor data sets, but whose volumes are known. The aim is to identify values of k_1 and k_2 in the weight function such that the resulting estimated tumor volumes will be as close to the known values as possible. We therefore created spherical tumor sets (radius 10 pixels) whose tumor intensities had means and standard deviations similar to those in the 10 real data sets shown in Table 1. Recognizing that the gradient is likely highest at the edge of the tumor, we set for k_2 the biweight mean of the gradient magnitude distribution. We

then calculated volumes for these tumors, using our marching cubes algorithm (see next section), as a function of k_1 , and identified the value of k_1 that gave a volume closest to the given volume of the simulated tumor. These values are also given in Table 1. Looking for some consistent metrics, we found that in each of the 10 data sets, the ratio of the number of Hounsfield units between the tumor mean and k_1 , to the number of Hounsfield units between the tumor mean and k_2 , is 0.465 (SD = 0.001). Hence, we choose for $k_1 = 0.535T_{\text{tumor}} - 0.465k_2$, where T_{tumor} is the estimated mean of the tumor intensity distribution from the EM algorithm.

Table 1: Means and standard deviations (SD) of tumor intensity distributions and edge gradient magnitude distributions from 10 clinical tumor data sets. Using simulated spherical tumors (radii = 10 pixels) with intensity means and SDs similar to those in columns 2 and 3, and setting k_2 equal to the biweight mean of the gradient distribution, k_1 (column 6) yields an estimated tumor volume closest to the actual volume for simulated data sets based on these 10 real data sets. Column 7 gives the ratio: $(\text{tumor mean} - k_1)/(\text{tumor mean} - k_2)$; mean = 0.465; SD = 0.001.

Data set	Tumor mean	Tumor SD	Gradient mean	Gradient SD	round(k_1)	Ratio *
S59	28.77	50.10	-441.86	171.828	-223	0.464
S72	22.92	72.10	-456.65	182.207	-234	0.465
S64	25.40	93.74	-474.97	188.819	-243	0.464
S52	23.93	88.46	-480.63	182.568	-246	0.465
S55	5.01	112.24	-501.62	189.524	-279	0.466
S66	-23.26	184.63	-491.83	168.978	-261	0.465
R23-1	20.43	90.59	-469.20	188.550	-243	0.462
R23-2	69.55	146.47	-473.28	179.743	-220	0.466
R32-1	18.05	112.75	-452.63	171.370	-234	0.465
R32-2	55.57	248.83	-483.40	176.912	-232	0.467

5 Volume Calculations

The shape of the lung tumor can be seen by visualizing a surface at a specified pixel intensity. We start by assuming that every pixel inside the surface having an isovalue $\text{round}(k_1)$ corresponds to tumor (versus background or edge). We calculate the volume of the enclosed isosurface as a starting point for our tumor volume calculation. This volume is computed by summing the volumes in each individual voxel that contains the tumor, using a marching cubes-type algorithm [12], in which each partially filled voxel geometry is computed based on a linear interpolation within that voxel. This algorithm is illustrated in Figure 5, which shows a partial volume as a set of tetrahedrons whose volumes can be calculated geometrically.

For each pixel value between $\text{round}(k1)$, the lowest value at the which the weight equals 1.0, and $\text{round}(k2)$, the highest value for which the weight equals 0.0, we calculate an

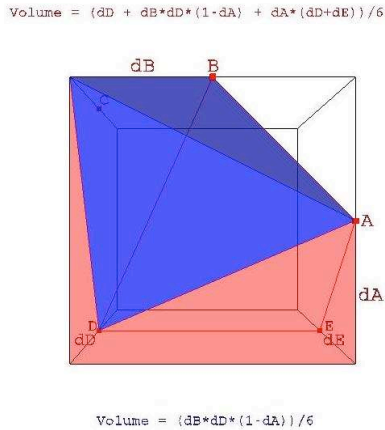


Figure 5: Example volume broken into sets of tetrahedra.

additional volume measurement for the pixels enclosed within that isosurface. In general, this set of isosurfaces produces a corresponding set of volumes, each slightly larger than the next. Each additional volume shell is multiplied by its corresponding weight and added to the total volume calculation. If we determine that a value of -150 HU, for example, is the lowest value with a probability of 1.0, we calculate an initial estimate of the tumor volume, V_0 , by calculating the volume enclosed by the corresponding isosurface and weight it by w_0 . Subsequent volumes, V_i , are calculated for each i less than $\text{round}(k2)$. With each subsequent calculation, we multiply the difference between it and the previous volume by w_i (see Section 3); i.e., $w_i(V_i - V_{i-1})$. We continue this process until we reach a volume whose difference from the previous volume is weighted by $w_i = 0$. The total volume of tumor is then calculated as a function of the individual volume calculations at each of the isovalues ($V_i = \text{volume at isovalue } i$) and their corresponding weights ($w_i = \text{weight at isovalue } i$) as:

$$V^* = V_0 + \sum_{i=1}^n w_i (V_i - V_{i-1}) \quad (3)$$

6 Uncertainties

Several sources contribute to the uncertainties in these calculations. Probably the largest source of error is due to the linear interpolation used in the voxel by voxel volume calculation. For tumors with large curvatures along their edges, especially very small tumors with large surface to volume ratios, these errors can be non-negligible. We created 3D gridded data to represent perfect spheres of different sizes, and computed their volumes using our marching cubes algorithm. With a 3D grid with equal interpixel distances, the systematic error due to the marching cubes algorithm varied from 2.37 % (for spheres of radius 5) to 0.15 % (for spheres of

radius 20). These errors rose significantly, however, when we used grids that more closely reflect the grids of the lung data; that is, the interpixel distances in one direction (in the direction of the CT slice) are larger than the those in the other directions (within one slice of CT data). Table 2 compares the volumes calculated on an even grid with volumes calculated on a grid resembling the data of Figure 1, in which the slice thickness is 1.25 mm and the distance between pixels within a slice is 0.57 mm. Although the marching cubes algorithm for computing volumes provides reasonably accurate estimates for data on a regular grid, the linear approximations in the non-regular grid lead to large errors, especially for small spheres (tumors). Currently, we are implementing a second order approximation in the slice direction to reduce these errors.

Table 2: Analytic vs. Numerical Volumes of Spheres in Lung Data

Radius	Actual volume	Volume 1 x 1 x 1 grid	%error 1 x 1 x 1 grid	Volume 1 x 1 x 2.9 grid	% error 1 x 1 x 2.9 grid
5	523.60	511.19	2.37	375.13	28.36
10	4188.79	4164.12	0.59	3571.21	14.74
15	14137.16	14099.81	0.26	12673.75	10.35
20	33510.29	33460.98	0.15	30920.93	7.73

A second source of uncertainties arises from the statistical method to assign weights. Each V_i involves uncertainty, which are combined to yield an overall uncertainty. Since the volume calculation is a weighted linear combination of V_i , the variance is, to a first approximation,

$$\text{Var}(V^*) = \sum_{i=1}^n w_i^2 \text{Var}(V_i) \quad (4)$$

The calculation is conservative because (a) w_i has uncertainty as well, due to the uncertainty in T and S that are used in it; and (b) the V_i are not uncorrelated. The effects of (a) and (b) tend to offset each other, so this estimate of the variance of the final calculation is adequate for most purposes. A bootstrap calculation verifies this claim, as will be shown in future work.

7 Comparisons With Other Approaches

In this section, we briefly compare this approach for lung tumor volume calculation to other types of approaches, since different approaches are useful for different overall goals. The method described here is focused on accurately and automatically measuring lung tumor volumes with measurable accuracy. Other methods may be better suited for other purposes. Thresholding methods can produce good results, which improve with the number of threshold values used [13], but assumes that tumor edges have consistent pixel intensities. Region-growing segmentation methods begin growth from a seed point of a connected region bounded by a threshold value

[14] and require user-specified values. Watershed segmentation, another region-based method that has been widely used in the medical community [15,16], also produces user-dependent results, as do other methods such as livewire segmentation, which utilizes gradients of the pixel field and can be very useful on images where steep gradients exist, such as the brain or liver [17]. In the method presented here, two different users will arrive at exactly the same volume for an object, an answer that is not influenced by various parameters input by the users.

8 Conclusions

We have described a method to calculate the partial volumes of voxels in medical lung CT images. This method is based on the assumptions that the pixel intensities defining the edges of an object form a separate, widely spread distribution from the pixel intensities in the center of the object. We propose an algorithm that assigns a weight to each pixel intensity that may be viewed as a "likelihood" that an individual pixel is a part of the tumor. This method uses the observed pixel distributions for tumor, background and edge pixel intensities, and the magnitude of the gradient of the pixel intensity to evaluate the parameters needed for volume measurement, k_1 and k_2 , thus requiring no user input parameters. The growth of tumors can be measured by comparing data that is taken at different times and possibly on different CT machines, since each data set is evaluated independently. Initial studies with measurements of synthetic tumors of known size show that we are able to measure simple objects with a fair amount of accuracy.

ACKNOWLEDGEMENTS

Support for Dr. Kafadar from the Army Research Office, Grant Number W911NF-05-1-0490 awarded to the University of Colorado-Denver, and from the National Science Foundation, Grant Number DMS0802295 awarded to Indiana University, is gratefully acknowledged

9 References

[1] Jirapatnakul, A.C., Reeves, A.P., Apanasovich, T.V., Biancardi, A.M., Yankelevitz, D.F., Henschke, C.I. Pulmonary Nodule Classification: Size Distribution Issues. *ISBI 2007*, 1248-1251.

[2] Kostis, W.J., Reeves, A.P., Yankelevitz, D.F., Henschke, C.I. Three-Dimensional Segmentation and Growth-Rate Estimation of Small Pulmonary Nodules in Helical CT Images. *IEEE Trans. on Medical Imaging* **22**, No. 10, October 2003.

[3] Bonnet, N., Cutrona, J., Herbin, M., A no-threshold histogram-based image segmentation method. *Pattern Recognition* **35**: 2319-2322 (2002).

[4] Reeves, A.P., Chan, A. B., Yankelevitz, D.F., Henschke, Kressler, C.I.B., Kostis, W.J. On measuring the change in size of pulmonary nodules. *IEEE Trans. on Medical Imaging* **25**(4):435-450 (2006).

[5] Preim, B., Bartz, D. Image Analysis for Medical Visualization. *Visualization in Medicine* 83-131, 2007.

[6] Hastie, T.; Tibshirani, R.; Friedman, J.H. *The Elements of Statistical Learning: Data Mining, Inference, And Prediction*. New York: Springer (2001).

[7] Kafadar, K. The Efficiency of the Biweight as a Robust Estimator of Location. *J. of Research of the National Bureau of Standards* **88**, No. 2, March-April 1983.

[8] Mosteller, F.; Tukey, J.W. *Data Analysis and Regression: A Second Course in Statistics*. Addison-Wesley: Reading, Massachusetts, 1977.

[9] Hoaglin, D.C.; Mosteller, F.; Tukey, J.W. *Understanding Robust and Exploratory Data Analysis*. Wiley: New York, 1983.

[10] Kafadar, K. A biweight approach to the one-sample problem. *J. Am. Statistical Assoc.* **77**:416-424 (1982).

[11] Peskin, A.P., Kafadar, K., Dima, A., Bernal, J., Gilsinn, D., Synthetic Lung Tumor Data Sets for Comparison of Volumetric Algorithms. *2009 International Conference on Image Processing, Computer Vision, and Pattern Recognition*.

[12] Lorensen, W.E., Cline, H.E. Marching Cubes: A High Resolution 3D Surface Construction Algorithm. *Computer Graphics* **21**, No. 4: 163-169 (1987).

[13] Haidekker, M.A., Andresen, R., Evertsz, C.J.G., Banzer, D., Peitgen, H.O. Issues of threshold selection when determining the fractal dimension in HRCT slices of lumbar vertebrae. *J Radiol* **73**: 69-72 (2000).

[14] Selle, D., Preim, A.S., Schenk, A., Peitgen, H.O. Analysis of vasculature for liver surgery planning. *IEEE Transactions on Medical Imaging* **21**(11): 1344-1357 (2002).

[15] Serra, J. *Image Analysis and Mathematical Morphology*. London: Academic Press, 1982.

[17] Hahn, J.K., Peitgen, H.O.: The Skull Stripping Problem in MRI Solved by a Single 3D Watershed Transform. *MICCAI* **1935**:134-143. Springer, Oct 2000.

[17] Mortensen, E.N.; Morese, B.; Barrett, W.A.; Upuda, J.K. Adaptive boundary detection using livewire two-dimensional dynamic programming. *IEEE Computers in Cardiology*, 635-638, 1992.

[18] Scott, D.W: Parametric Statistical Modeling by Minimum Integrated Square Error, *Technometrics* **43**:274-285, 2001.

Appendix 1: EM algorithm for 3-component Gaussian mixture

The model for an observation X that comes from a three component Gaussian mixture is

$$X = a_1 X_1 + a_2 X_2 + a_3 X_3 \quad (4)$$

where X_j comes from a Gaussian distribution with mean μ_j and variance σ_j^2 (standard deviation σ_j), a_1, a_2 , and $a_3 = 1 - a_1 - a_2$ take values in $\{0,1\}$, and $P\{a_i = 1\} = \pi_j$, $j = 1, 2, 3$. That is, (a_1, a_2, a_3) has a multinomial(3) distribution with parameters π_1 and π_2 ($\pi_3 \equiv 1 - \pi_1 - \pi_2$). The probability density function (pdf) of this three-component Gaussian mixture is

$$f(x; \theta) = \pi_1 \phi_{\mu_1, \sigma_1}(x) + \pi_2 \phi_{\mu_2, \sigma_2}(x) + \pi_3 \phi_{\mu_3, \sigma_3}(x) \quad (5)$$

where $\pi_j \phi_{\mu_j, \sigma_j}(x) = (\sqrt{2\pi} \sigma_j)^{-1} \exp[-(x - \mu_j)^2 / (2\sigma_j^2)]$ denotes the Gaussian pdf with mean μ_j and variance σ_j^2 (standard deviation σ_j), π_j denotes the proportion of pixels in component j ($\pi_3 \equiv 1 - \pi_1 - \pi_2$), and θ denotes the vector of parameters $(\pi_1, \pi_2, \pi_3, \mu_1, \mu_2, \mu_3, \sigma_1^2, \sigma_2^2, \sigma_3^2)$. To fit these 8 parameters using the EM (expectation-maximization) algorithm using data values $\{x_1, \dots, x_n\}$, we start by recognizing that if we know for each observation the population from which it comes, then we would use the observations from each population to fit the Gaussian means and variances in the usual way using sample means and sample variances:

$$\hat{\mu}_j = \sum_{i \in S_j} x_i / n_j, \quad \hat{\sigma}_j^2 = \sum_{i \in S_j} (x_i - \hat{\mu}_j)^2 / n_j \quad (6)$$

where S_j is the set of indices of x_i which come from population j and n_j is the number of indices in S_j . Since we do not know a priori the population from each x_i comes, we proceed as follows. (See also Hastie, Tibshirani, and Friedman 2001, Section 8.5, pp. 236–243. The EM algorithm for the 2-component Gaussian mixture appears on p.238.)

1. Initialization step: Start with initial estimates of θ ; e.g.,

$$\begin{aligned} \mu_1 &= -800, \quad \sigma_1 = 100 \text{ (background pixels)} \\ \mu_2 &= -400, \quad \sigma_2 = 200 \text{ (edge pixels)} \\ \mu_3 &= 50, \quad \sigma_3 = 100 \text{ (foreground pixels)} \end{aligned}$$

The algorithm converges more quickly with closer start values but precise start values are not critical.

2. Expectation step: Compute the “indicators” for each observation x_i , corresponding roughly to the probabilities that a given observation x_i comes from each of the populations:

$$\begin{aligned} w_{i1} &= \hat{\pi}_1 \phi_{\hat{\mu}_1, \hat{\sigma}_1}(x_i) / D \\ w_{i2} &= \hat{\pi}_2 \phi_{\hat{\mu}_2, \hat{\sigma}_2}(x_i) / D \\ w_{i3} &= 1 - w_{i1} - w_{i2} = (1 - \hat{\pi}_1 - \hat{\pi}_2) \phi_{\hat{\mu}_3, \hat{\sigma}_3}(x_i) / D \\ D &\equiv \hat{\pi}_1 \phi_{\hat{\mu}_1, \hat{\sigma}_1}(x_i) + \hat{\pi}_2 \phi_{\hat{\mu}_2, \hat{\sigma}_2}(x_i) + (1 - \hat{\pi}_1 - \hat{\pi}_2) \phi_{\hat{\mu}_3, \hat{\sigma}_3}(x_i) \end{aligned} \quad (7)$$

3. Maximization (updating) step: We now treat the “indicators” as weights and compute the maximumlikelihood estimators of the Gaussian means and variances in the usual fashion, for $j = 1, 2, 3$:

$$\begin{aligned} \hat{\mu}_j &= \sum_{i=1}^n w_{ij} x_i / \sum_{i=1}^n w_{ij} \\ \hat{\sigma}_j^2 &= \sum_{i=1}^n w_{ij} (x_i - \hat{\mu}_j)^2 / \sum_{i=1}^n w_{ij} \end{aligned} \quad (8)$$

and update the estimated proportions in the three components as

$$\hat{\pi}_j = \sum_{i=1}^n w_{ij} / n \equiv w_{+j} / n \quad (9)$$

4. Iteration step: Repeat steps 2 and 3 until convergence criterion is reached.

We note the following:

- Scott[18] proposed an algorithm for fitting Gaussian mixtures based on a different criterion for estimation. The algorithm has advantages particularly in robustness (insensitivity to possible outliers in the data). We expected rather few outliers in these data but are planning future work to investigate Scott’s algorithm on data such as those considered here.