An Estimation/Correction Algorithm for Detecting Bone Edges in CT Images

W. Yao, P. Abolmaesumi, *Member, IEEE*, M. Greenspan, *Member, IEEE*, and R. E. Ellis*, *Member, IEEE*

***Abstract*—The normal direction of the bone contour in computed tomography (CT) images provides important anatomical information and can guide segmentation algorithms. Since various bones in CT images have different sizes, and the intensity values of bone pixels are generally nonuniform and noisy, estimation of the normal direction using a single scale is not reliable. We propose a multiscale approach to estimate the normal direction of bone edges. The reliability of the estimation is calculated from the estimated results and, after re-scaling, the reliability is used to further correct the normal direction. The optimal scale at each point is obtained while estimating the normal direction; this scale is then used in a simple edge detector. Our experimental results have shown that use of this estimated/corrected normal direction improves the segmentation quality by decreasing the number of unexpected edges and discontinuities (gaps) of real contours. The corrected normal direction could also be used in postprocessing to delete false edges. Our segmentation algorithm is automatic, and its performance is evaluated on CT images of the human pelvis, leg, and wrist.**

***Index Terms*—Edge-tracing, image segmentation, Kalman filtering, normal direction, postprocessing, reliability, seed.**

I. Introduction

SEGMENTATION of computed tomography (CT) images is an important step in image-guided surgery. In orthopedic applications, the segmentation of bone from soft tissue is critical in both the preoperative planning phase and the intraoperative registration phase. Cortical bone, which is both physically and radiologically dense, is easy to segment from the surrounding soft tissues; however, cancellous bone is physically much less dense and radiologically can appear to be similar to the surrounding tissues. This is particularly true for images of elderly patients, whose decreased bone density and increased calcification of muscle and vascular tissues further complicates the segmentation process.

Several classes of methods have been proposed to segment bones from CT images in the literature [1]–[6]. Global [7], [8] or local [9] thresholding approaches assume the homogeneity of the objects being segmented. While this assumption usually holds for fat and often for muscle, it does not usually hold for bone. Bones have different shapes and rich inner structures, which cause the intensity of bone pixels to vary in a wide range from the intensity of fat to the highest value of the image. Since the intensity of bone overlaps with the intensity of muscle and fat, it is difficult to segment bone from the image by using an intensity threshold.

Active contour models, or snakes, were introduced by Kass *et al.* [2] and have been shown to be very useful in medical image processing [10], [11]. An active contour evolves under the control of so-called energy terms that describe relevant properties of the contour, such as smoothness, continuity, and an external "force" coming from the nonuniform intensity distribution of the image. The performance of active contour models depends critically on the initial contour [12], which can be provided manually or automatically (e.g., by using information such as an anatomical atlas [13]). Roughly speaking, if the initial contour is close to the real contour of a bone, the active contour usually converges to the real contour. However, such an initial contour is hard to automatically obtain and, when the initial contour is not good enough, the contour may not converge to the real one.[1]

Recently, level sets [14]–[16] have shown promise in segmentation of medical images, such as CT slices of human brains. Unlike snakes, which express the driven terms as energy and seek the minimal energy state, level sets consider an implicit surface driven by advection, propagation and some spatial modifier terms, and take the stable zero level set as the contour of the segmented object. In Section III-C, we evaluate a level set method for segmenting bones.

Other methods that have been used in bone segmentation include region growing [5], [17], region competition [18], watershed segmentation [19], and skeletally coupled deformable models [20]. There are excellent recent reviews of these powerful methods in the literature [12], [20].

Edge detection, or edge filtering, is another class of methods that can be used to find bone contours in CT slices when the intensity of pixels near the boundary changes more rapidly than that in the nonboundary region. Edge filters often have one or more adjustable parameters, and the multiscale property of images [1], [21] can make it difficult to obtain the optimal values of these parameters. Edge filters often produce nonclosed contours or closed contours with many false edges; Fig. 1, for example, shows that the Canny edge filter [22] does not perform well even

Manuscript received August 20, 2004; revised April 4, 2005. This work was supported in part by the Natural Sciences and Engineering Research Council of Canada, in part by the Institute for Robotics and Intelligent Systems, and in part by the Ontario Research and Development Challenge Fund. The Associate Editor responsible for coordinating the review of this paper and recommending its publication was M. Loew. *Asterisk indicates corresponding author.*

W. Yao was with the School of Computing, Queen's University, Kingston, ON, Canada. He is now with the Department of Applied Mathematics, University of Western Ontario, London, ON, Canada (e-mail: wyao4@uwo.ca).

P. Abolmaesumi is with the School of Computing, Queen's University, Kingston, ON, Canada (e-mail: purang@cs.queensu.ca).

M. Greenspan is with the Department of Electrical and Computer Engineering, Queen's University, Kingston, ON, Canada (e-mail: michael.greenspan@ece.queensu.ca).

*R. E. Ellis is with the School of Computing, Queen's University, Kingston, ON, Canada (e-mail: ellis@cs.queensu.ca).

Digital Object Identifier 10.1109/TMI.2005.850541

[1] We provide examples of good and bad convergence in Section III-B.

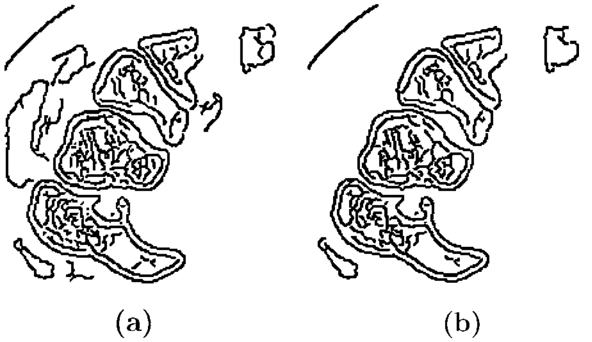

Fig. 1. Edge maps obtained by the Canny detector from the CT image shown in Fig. 10(a) when the variance of the Gaussian is 1.2, the low-end hysteresis threshold is 20%, and the high-end hysteresis threshold is, respectively, 70% in (a), and 75% in (b). The source code is from http://figment.csee.usf.edu/sarkar/resume/default.htm.

though the bone contour is physically closed and can be mathematically modeled as closed [23]. Furthermore, in bone segmentation only the edges of the bone tissue are desired so the edges of other tissues such as fat and muscle should not appear in the resulting edge map. Additional techniques are, therefore, needed to form a closed contour and to suppress or delete undesired edges.

An *edge-tracing* or *contour-tracing* algorithm can be used to form a closed contour [24]. Let us say that the true location of the edge point $i+1$ is $\mathbf{x}_{i+1}$. Starting from the ith detected edge point (or seed) at $\mathbf{x}_i \in \mathbf{R}^2$, the next edge point at $\widetilde{\mathbf{x}}_{i+1} = f(\mathbf{x}_i)$ can be expressed as

$$\mathbf{x}_{i+1} = \widetilde{\mathbf{x}}_{i+1} + \epsilon_{i+1} \tag{1}$$

where $f(\mathbf{x}_i)$ is a function to predict $\mathbf{x}_{i+1}$ from $\mathbf{x}_i$ and ϵ_{i+1} is the error of the predictor as applied to the given image. When ϵ_{i+1} is estimated, the edge-tracing process continues to obtain the next point until the previously predicted point is revisited or certain conditions are no longer satisfied.

Correcting the prediction, or equivalently finding ϵ_{i+1}, involves the use of edge detectors such as the first derivative of the Gaussian, Canny's detector [22], edge detectors based on zero-crossings [25] or optimal linear filters [26]. These detectors are designed for one-dimensional (1-D) noisy step-edge signals. When they are used to detect edges in two-dimensional (2-D) images, optimally the 1-D signal should consist of the intensity of pixels on the normal line of the predicted point since the intensity changes the most rapidly along the normal direction. Thus, a critical step is to accurately determine the normal direction at the predicted point.

II. Our Estimation/Correction Algorithm

A. Description

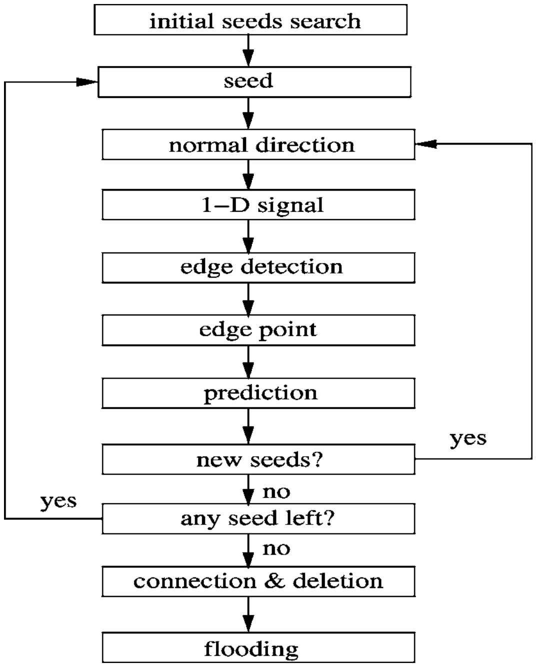

Fig. 2. Block diagram of our segmentation algorithm.

Our algorithm, based on edge tracing, is depicted as a block diagram in Fig. 2. Suppose we have initial seeds obtained from some method such as thresholding or manual selection. The seeds do not need to fall directly on the edge of the bone (but should be near the edge). Starting from an initial seed, to find an edge point near the seed or to correct the location of the seed we calculate the normal direction at the seed and construct a 1-D signal that consists of the intensity of the pixels along the normal direction centered at the seed. We then use an edge filter to find the edge point in the 1-D signal. The deviation of the edge point in the 1-D signal from the center is used to obtain the location of the edge point in the 2-D image. From the edge point, we predict the next edge point along the edge. The predicted point is taken as a seed and the process is repeated until no new edge point can be predicted. We then go to the next initial seed, and the entire process is repeated until all of the initial seeds are used up. Finally, the edge map is postprocessed to produce a better edge map of the bone.

Ideally, bone contours are closed. However, due to noise, the contour resulting from the above described process may not be closed and some new information or assumption is needed to enclose the contour. One common approach is Canny's *hysteresis thresholding* [22], which uses two separate threshold values. Those pixels with an edge strength greater than the larger threshold value are deemed to be edge pixels, while those with an edge strength less than the lower threshold value are not on an edge. Those pixels with intermediate edge strength, between the smaller and larger thresholds, are determined to also be at edge only if they are the neighbor of an edge pixel. In our algorithm, instead of hysteresis thresholding, we use the smoothed 1-D signal along the tangent direction of the edge so that the neighbors' edge strength could influence the edge strength of the predicted point. The difference in these two methods is that the latter uses more than one neighbor's information, whereas the former uses the information of the nearest neighbor only. The cost of the latter is that the tangent direction or the normal direction needs to be estimated more accurately than in Canny's method.

There are some obvious advantages to using the normal direction in image processing. First, the normal direction near an edge is more stable than the intensity of pixels near the edge;

we have observed that the direction at one pixel can stay unchanged with different scales of observation (see Section II-B for details). Assuming that edges are smooth, the difference between the normal directions at two neighboring pixels along an edge cannot be large. Second, the normal direction encodes geometrical information of the edge, which is very useful in image processing. For example, in the popular snakes method of segmentation, there is an energy term for the smoothness of the edge [2]. As we will show below, these properties of the normal direction can be utilized to solve some challenging problems in image processing.

Due to the noise and multiple scales of the image, accurate edge-normal direction is not easy to obtain, which is one of the reasons why edge tracing is not robust in producing true edges. In this paper, we investigate this issue and design our segmentation algorithm based on a consideration of the normal direction. In Section II-B, we use the first derivative of the Gaussian as the normal direction estimator to show the sensitivity of the estimated normal direction on the variance σ, the parameter of the estimator, and on the coordinate system. We show how to correct the estimated direction under the assumption of the smoothness of the edge in Section II-C. We discuss the reliability of the estimated normal direction in Section II-D, and show how reliability can be rescaled to make our algorithm robust and how an optimal value of σ can be determined. In Section II-E we briefly discuss the construction of a 1-D signal where the neighboring 1-D signals are adopted to smooth the current 1-D signal. Section II-F discusses the search of initial seeds so that our segmentation algorithm can be completely automatic. Because not all initial seeds determined in Section II-F are near the edge of bone, some false edges are unavoidable; a solution to this problem is presented in Section II-G, where we delete the false edges between bones by classifying the neighboring detected edge points into three groups according to their normal directions.

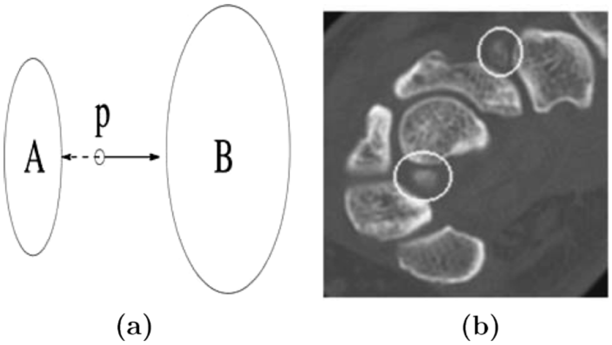

Fig. 3. (a) The error of the estimated normal directions at point p can be as large as 180° depending on the value of σ and the intensity of objects A and B. (b) A CT slice of human wrist in which the marked regions show the situation discussed in (a).

B. Normal-Direction Estimation

Our normal-direction estimation is based on the observation that the intensity of pixels near an edge changes more rapidly along the normal direction than along the tangent direction. Noise and quantization of an underlying smooth signal, which are unavoidable in CT images, can blur or otherwise obscure the edge. Therefore, a good edge normal direction estimator should decrease the effect of noise. Approximating the noise in images as additive Gaussian noise [27], G_n, the first derivative of the 2-D symmetrical Gaussian is widely used to obtain the normal direction, and to suppress the noise at the same time

$$G_n = \frac{\partial G}{\partial \mathbf{n}} = \mathbf{n} \cdot \nabla \mathbf{G} \tag{2}$$

$$G(\mathbf{x}) = \exp\left(-\frac{\|\mathbf{x}\|^2}{2\sigma^2}\right) \tag{3}$$

where $\mathbf{x} \in \mathbf{R}^2$, σ is an estimate of the variance of the noise. The normal direction $\mathbf{n}$ at $\mathbf{x}_0$ can then be estimated as

$$\mathbf{n}(\mathbf{x}_0) = \frac{\nabla\left(\mathbf{G}(\mathbf{x}-\mathbf{x}_0) * \mathbf{I}\right)}{\|\nabla\left(\mathbf{G}(\mathbf{x}-\mathbf{x}_0) * \mathbf{I}\right)\|} \tag{4}$$

where $*$ denotes convolution, and $\mathbf{I}$ is the image. It is important to note that two assumptions underly the use of (4) to estimate the normal direction.

First, it is assumed that the value of σ can be determined by some method. When the image consists of step edges and additive Gaussian noise, σ is the variance of the noise and can be estimated by the autocorrelation function of the image [22]. Modeling a real image in this way, however, is a poor approximation because the autocorrelation function is notoriously inaccurate. Instead, one often has to experiment with many values of σ to find the best value for a specific image. In image-guided surgery, our CT scans typically have hundreds of slices so we require that the optimal value of σ for each slice must be determined automatically.

Unfortunately, the normal direction estimated by (4) can be sensitive to the choice of σ, especially when $\mathbf{x}_0$ is between two nearby bones. For example, in Fig. 3(a) we calculate the normal direction at point p. It is expected that the direction should be along the solid line because p is closer to bone A. However, if the intensity of bone A is less than the intensity of bone B, and the value of σ is not small enough, the estimated normal direction could be along the dashed line as shown in Fig. 3(a). These cases often happen in our medical images, such as those of the human wrist which has 7 small carpal bones. As shown in Fig. 3(b), in some CT slices there is only one-pixel or two-pixel space between the bones; furthermore, some bones are radiologically dense whereas others (with more cancellous bone and less cortical bone) are imaged with lower intensity values. In this case, the estimated normal direction is highly sensitive to σ in the marked region. Note that the circled regions are bone ends, where a three-dimensional (3-D) extension of our 2-D algorithm would likely have improved performance.

The second assumption in estimating the normal direction is that the result from (4) will not change under a coordinate system rotation. As stated above, the normal direction estimation is based on the fact that the intensity value information changes most rapidly along the normal direction but before evaluating (4), one does not know the normal direction so one must calculate the normal direction in the original coordinate system. One can then check the result in a new coordinate system whose axes are parallel and vertical to the estimated normal direction,

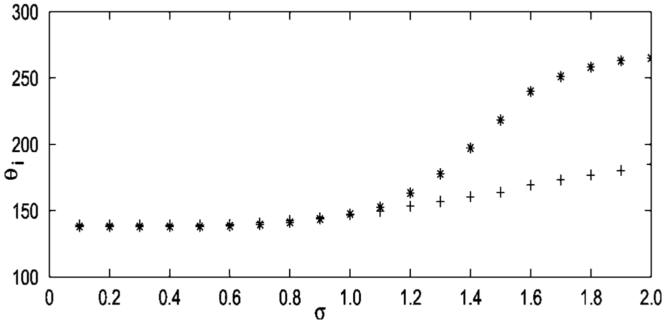

Fig. 4. The estimated normal directions under various σ and with or without coordinate system rotation. "+" denotes the normal directions estimated in the original coordinate system, and "*" denotes those in the coordinate system rotated 45°.

respectively. Repeatedly performing (4) in the updated coordinate system, one may expect that the estimated normal directions are distributed around some value with small discrepancy. However, our earlier work showed that the estimated normal directions can be very different when the coordinate system rotates even slightly [28] and that normal directions may oscillate periodically or quasiperiodically with large magnitudes. One reason for the estimation difference is the quantization of the pixel size but, even if we use the original coordinate system and a 45° rotated coordinate system so that there is no effect from quantization, we found that the estimated normal directions could vary greatly. Fig. 4 shows one example. The angle θ_i between the normal direction at a point and the horizontal coordinate (x) is estimated in two coordinate systems with various σ. The point is one pixel away from the edge of a bone. From the figure, it can be seen that the estimated normal directions are very different in the two coordinate systems for some values of σ. This is because the convolution region of the image is different under different coordinate systems and the intensity changes nonlinearly.

Estimating the normal direction in noisy quantized images, therefore, remains a challenge. Some researchers estimate the edge strength in rotated coordinate systems [29]. The result, which in our case is the average normal direction case, may be better than a single estimate but is costly to compute because the effect of quantization must be ameliorated. To avoid large errors in the estimation of the normal direction of an edge, we propose the use of *geometrical* information to improve the results.

C. Normal-Direction Correction

Because of the difficulty to accurately estimate the normal direction, we consider a predictor-corrector scheme to improve the estimation. Compared to the intensity of pixels near an edge, the geometrical structure of the edge of bone may turn out to be more predictable than intuition first suggests. We will assume that the edge of bone is smooth and continuous so that, if we have some detected edge points, we can predict not only $\widetilde{\mathbf{x}}_{i+1}$ in (1) but the normal direction at $\widetilde{\mathbf{x}}_{i+1}$ as well.

Assume that the edge of bone is smooth and C^2 continuous. It then follows that the normal direction $\mathbf{n} \subset C^1$ because the normal direction is the first derivative of the edge. Different requirements on image smoothness have been proposed; for example, Torre and Poggio [30] required the image to be C^∞ and Dobkin *et al.* [24] required the image to be C^1. We require that the image be C^2 because we need the existence of the second derivative of the normal direction along the edge. Let θ_i denote the angle spanned by the normal direction at pixel i in the horizontal coordinate, and let $\Delta\theta_i$ denote the angle spanned by the normal directions at two neighboring edge pixels $i-1$ and i. The difference between $\Delta\theta_{i-1}$ and $\Delta\theta_i$ should be small. When the contour of the bone is a circle, ideally the difference

$$\Delta^2\theta_i = \Delta\theta_i - \Delta\theta_{i-1} = 0. \tag{5}$$

In practice, this will seldom be true because the contour is seldom a circle; also, the normal directions may not be estimated accurately. Instead, we assume that $\Delta^2\theta_i$ is randomly distributed around zero, so

$$\Delta^2\theta_i = \xi \tag{6}$$

where ξ is a random variable. The expected value $\langle\xi\rangle = 0$, but the variance may in practice be large.

Estimation error can cause a large variance of ξ. To correct the estimation, or equivalently to decrease the variance, we multiply both sides of (6) by a factor $q_i \in [0, 1]$. Denote by $\bar{\theta}_i$ the corrected normal direction at pixel i

$$\Delta^2\bar{\theta}_i \equiv q_i\Delta^2\theta_i. \tag{7}$$

For the left side of (7), similar to (5) we have $\Delta^2\bar{\theta}_i = \Delta\bar{\theta}_i - \Delta\bar{\theta}_{i-1}$. For the right side of (7), we further assume that the normal direction at pixel $i-1$ has been corrected when we estimate θ_i, then instead of (5) we have $\Delta^2\theta_i = \Delta\theta_i - \Delta\bar{\theta}_{i-1}$. Combining these results into (7), we have

$$\Delta\bar{\theta}_i - \Delta\bar{\theta}_{i-1} = q_i(\Delta\theta_i - \Delta\bar{\theta}_{i-1}). \tag{8}$$

And from (8), we have

$$\Delta\bar{\theta}_i = q_i\Delta\theta_i + (1 - q_i)\Delta\bar{\theta}_{i-1}. \tag{9}$$

We then rewrite (9) in the normal directions explicitly by using

$$\Delta\bar{\theta}_i = \bar{\theta}_i - \bar{\theta}_{i-1} \tag{10}$$

and

$$\Delta\theta_i = \theta_i - \bar{\theta}_{i-1} \tag{11}$$

we obtain

$$\bar{\theta}_i = q_i\theta_i + (1 - q_i)(\bar{\theta}_{i-1} + \Delta\bar{\theta}_{i-1}) \tag{12}$$

where θ_i is the normal direction at pixel i estimated by (4), and $\bar{\theta}_{i-1} + \Delta\bar{\theta}_{i-1}$ is another normal direction at pixel i but estimated according to the edge smoothness assumption. The value of q_i can now be seen as the weight to balance these two estimates, or the *reliability* of the estimate based on (4). If we think θ_i is completely reliable, we take $q_i = 1$; if θ_i is completely unreliable, we take $q_i = 0$, and we use $\bar{\theta}_{i-1} + \Delta\bar{\theta}_{i-1}$ as the normal direction at pixel i. In general, $q_i \in [0, 1]$.

Equation (12) is equivalent to a 1-D Kalman filter [31] for modeling constant acceleration processes. The normal direction in our case is the state in the Kalman filter. The Kalman filter has been applied in many predictor-corrector problems [32], [33] where neither the measurement nor the estimation is error-free. To determine parameter q_i in Kalman filter, one needs to know the variances of the measurement and estimate errors *prior to* the correction. In our edge detection problem, however, these variances are hard to estimate because before the detection we do not know whether or not there is an edge and the variances could change hugely. In the next section, we introduce our scheme to estimate q_i.

To see why we can use (12) to correct the estimated normal direction, consider (9) when q_i is constant (in practice, q_i usually changes for each pixel). If q_i is a constant, $q_i = q$, from (9), we have after some trivial deduction

$$\Delta\bar{\theta}_i = q\sum_{j=0}^{n}(1-q)^j\Delta\theta_{i-j} + (1-q)^{n+1}\Delta\bar{\theta}_{i-n-1}. \quad (13)$$

When $n \rightarrow \infty$ and $q \in (0,1)$, the last term $(1-q)^{n+1}\Delta\bar{\theta}_{i-n-1} \rightarrow 0$. Equation (13) can then be written as

$$\Delta\bar{\theta}_i = q\sum_{j=0}^{\infty} f_j\Delta\theta_{i-j} \quad (14)$$

where $f_j = (1-q)^j$. Or, in continuous case

$$\begin{aligned}\Delta\bar{\theta}_i =& q\int_0^{\infty} f(x)\Delta\theta(x_i - x)dx\\ =& qf * \Delta\theta \end{aligned} \quad (15)$$

where $f(x) = (1-q)^x = \exp(-\alpha x)$, $\alpha = -\ln(1-q)$, $x \in [0,\infty)$. Thus $f(x)$ or f_j can be taken as a filter, *which is not a Gaussian distribution.*

A condition for filter $f(x)$ or f_j to work properly is that the distribution of $\Delta\theta$ must be similar to the filter. To test this in a single image, we first manually segmented bone from a CT slice of a human wrist and estimate the normal directions at the edge points by the method described in the next section. We then calculated the difference of the normal directions at neighboring edge points to obtain $\Delta\theta$. Fig. 5(a) displays the distribution of $\Delta\theta$ displayed as a solid line. For comparison purposes, the distributions of normalized $f(x) = (c/2)\exp(-c|x|)$ and the Gaussian $g(x) = \sqrt{1/2\pi\sigma^2}\exp(-x^2/2\sigma^2)$ are plotted with a dashed line and the symbol '+', respectively. It can be seen that the distribution of $\Delta\theta$ is much closer to $f(x)$ than $g(x)$, so filter f is likely to work properly to correct the normal direction in this image. Fig. 5(b) depicts the distribution of $\Delta\bar{\theta}$ corrected by using an optimal value of q_i at each edge point. It can be seen from the figure that there are more values of $\Delta\bar{\theta}$ nearby 0 and the variance of $\Delta\bar{\theta}$ is less than that of $\Delta\theta$; in the next section, we will discuss how to determine the optimal value of q_i.

When $q_i = q$, the filter $f(x)$ is the same as the Shen-Castan filter for mono-step edges [26] except that $x \in [0,\infty)$ because we only use the normal directions of detected edge points. So far our algorithm may be looked at as the generalized version of the Kalman filter and Shen-Castan filter. In practice, however, q_i is not constant so we propose a scheme to estimate q_i.

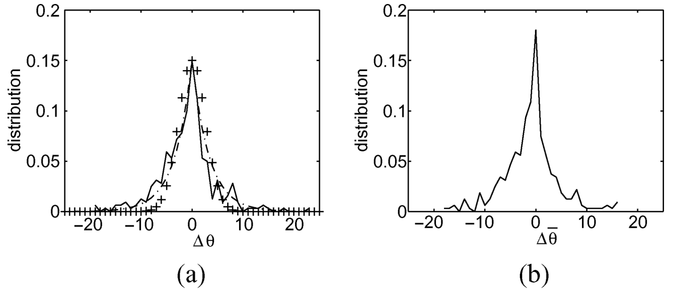

Fig. 5. Distributions of (a) estimated $\Delta\theta$ and (b) corrected $\Delta\bar{\theta}$, see text for details.

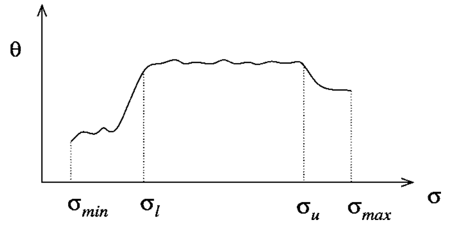

Fig. 6. Estimated normal direction θ versus the parameter σ of the normal direction estimator, the first derivative of the Gaussian.

D. Estimation of Reliability and Variance

Ideally, for a given edge in an image, the normal direction at one edge point should not change with different scales of observation. The estimated result from (4), however, changes with the scales represented by the variance σ. The reliability of the estimated normal direction can be calculated by observing this change.

Suppose pixel i is near an edge and the noise is small in its neighborhood when examined at a chosen scale. When we calculate the normal direction using (4) with different σ in a chosen interval $[\sigma_{\min}, \sigma_{\max}]$, if the result varies in a small range $\Delta\theta_{\text{tol}} > 0$ then we may say that the estimated result is reliable and set $q_i = 1$. Generally, we may define the reliability as

$$q_i = \frac{\sigma_u^i - \sigma_l^i}{\sigma_{\max} - \sigma_{\min}} \quad (16)$$

where $\sigma_u^i \leq \sigma_{\max}$ and $\sigma_l^i \geq \sigma_{\min}$ are the upper and lower bounds of the value of σ where the discrepancy of the estimated normal directions at pixel i is less than or equal to $\Delta\theta_{\text{tol}}$. Beyond these bounds, the discrepancy is greater than $\Delta\theta_{\text{tol}}$. In other words, we search for the θ which has the longest survival time in the series, and its value is allowed to oscillate with the tolerance $\Delta\theta_{\text{tol}}$. For example, consider the series of $\theta_i(\sigma)$ plotted in Fig. 6. We search the maximal region of σ, $[\sigma_l^i, \sigma_u^i]$, so that for each θ_i in this region the absolute value of the difference between θ_i and $\langle\theta_i^{l,u}\rangle$ is not greater than $\Delta\theta_{\text{tol}}$, where $\langle\theta_i^{l,u}\rangle$ is the average of $\theta_i(\sigma)$ when σ changes from σ_l^i to σ_u^i.

We take the value of $\langle\theta_i^{l,u}\rangle$ as the normal direction at pixel i. Our experiments show that such an automatically adjusted multiscale estimate is better than a single-scale estimate (as described in Section III). The value of $\sigma_{\min}$ can be chosen to be small (e.g., 0.1) so that the estimate responds to bones that are very close each other. The value of $\sigma_{\max}$ can be large (e.g., 2.0) to find edges of largely cancellous bones. In practice we recommend that the value of $\Delta\theta_{\text{tol}}$ be chosen carefully because it controls the sensitivity of the estimate. From our experiments we have determined that $\Delta\theta_{\text{tol}} = 10^\circ$ works well as a control parameter to obtain a better estimated normal direction.

The value of q_i determined by the above approach rely heavily on the values of $\sigma_{\min}$, $\sigma_{\max}$ and $\Delta\theta_{\text{tol}}$. For example, if $\Delta\theta_{\text{tol}}$ is too large, $\sigma_l^i \rightarrow \sigma_{\min}$ and $\sigma_u^i \rightarrow \sigma_{\max}$, namely, $q_i \rightarrow 1$ because even a large discrepancy of the estimated normal directions may be less than $\Delta\theta_{\text{tol}}$. On the contrary, if $\Delta\theta_{\text{tol}}$ is too small, $q_i \rightarrow 0$.

Therefore, an important step to obtain a reasonable reliability measure is to rescale q_i. That is, we need a re-scale factor γ so that γq_i is not so sensitive to the values of $\sigma_{\min}$, $\sigma_{\max}$ and $\Delta\theta_{\text{tol}}$. To do this, we first estimate the normal direction and q_i for each pixel of the slice (To save computation, we calculate the normal direction only at those possible edge points of bone. See Section II-F). We obtain a distribution of $\{q_i\}$ which depends on $\sigma_{\min}$, $\sigma_{\max}$ and $\Delta\theta_{\text{tol}}$. To re-scale $\{q_i\}$, we assume that the normal directions at 80% of the pixels are estimated by the approach described above with at least 90% reliability. The re-scale factor γ can then be obtained. This assumption is expected because we will use the normal direction to do the segmentation. Finally we use γq as the reliability. Equation (12) becomes

$$\bar{\theta}_i = \gamma q_i \theta_i + (1 - \gamma q_i)(\bar{\theta}_{i-1} + \Delta\bar{\theta}_{i-1}). \qquad (17)$$

During the normal-direction correction stage using (17) the value of γ is a constant. Thus, to obtain $\bar{\theta}_i$, we only need the information from pixel i (θ_i and q_i) and pixel $i-1$ ($\bar{\theta}_{i-1}$ and $\Delta\bar{\theta}_{i-1}$), although from (17) we are using information of all detected pixels along the edge. This is a distinct advantage of our approach.

Next, we can use the determined σ_l^i and σ_u^i to estimate the optimal σ at point i that will be used in the edge detector. The value of σ should lie within $[\sigma_l^i, \sigma_u^i]$. Because we are using a gradient-maximum edge detector, we choose the σ with the strongest edge strength $||\boldsymbol{\nabla} G(\sigma) * \mathbf{I}||$ as the optimal σ at point i, that is

$$\left\{\sigma \middle| \max\left\{||\nabla \mathbf{G}(\sigma) * \mathbf{I}||\right\},\ \sigma \in \left[\sigma_l^i, \sigma_u^i\right]\right\}. \qquad (18)$$

E. 1-D Signal Construction and Edge Detection

Once we have estimated and corrected the normal direction, and have obtained the optimal σ at pixel i, we can then construct a 1-D signal s_i composed by the intensity of pixels along the normal direction and centered at pixel i. To decrease the perturbations caused by noise and quantization effects we use a smoothed 1-D signal along the tangent direction of the edge. As stated above, we assume that the normal direction is estimated well so that the intensity of s_i changes most rapidly at the edge point. If the normal direction is not estimated well, the response of s_i to an edge filter is weaker, and the detected edge point is not as reliable. Therefore, similar to the estimated normal direction, the reconstructed s_i may not be completely reliable. Because s_i is determined from the normal direction, the reliability of s_i is the same as the reliability of the normal direction so the smoothed 1-D signal can be represented as

$$\bar{s}_i = \gamma q_i s_i + (1 - \gamma q_i)\bar{s}_{i-1}. \qquad (19)$$

Similar to our normal-direction correction, in smoothing the 1-D signal we only need the information from pixel i and $i-1$.

Generally, the reliability is low at noisy edge points. The strong nearby edge points can then have a bigger effect on the noisy edge points. By using the smoothed 1-D signal, the noise is depressed and the weak edge strength is enforced. The smoothed 1-D signal, thus, plays a role similar to Canny's hysteresis and is very useful in preventing or decreasing the possibility of *streaking*, which is a common problem in edge detection [22] that produces discontinuity in contours. One important goal of our edge-tracing algorithm is to minimize contour discontinuity.

In our experiments, we use finite 1-D $\boldsymbol{\nabla} G$, the first derivative of the Gaussian, as the edge detector. Besides its simplicity, it has been used as an approximation of the Canny edge detector [22], and much other work has demonstrated its advantages [29], [34]–[36]. In our case, by using the corrected normal direction, optimized σ, and smoothed 1-D signal, $\boldsymbol{\nabla} G$ performs very well indeed. $\boldsymbol{\nabla} G$ is a gradient maximum edge detector, and the pixel with the maximal edge strength in the smoothed 1-D signal is taken as an edge point.

Up to now in our presentation we have assumed that we have had a seed pixel i near the edge of a bone in a CT image. The initial seed i can be determined manually but, in order to perform the segmentation without manual intervention, we need a way to find initial seeds in each slice automatically.

F. Selecting Seed Points

How to find good seed points that are near bone edges is a complicated issue, especially when bones are close to each other and the edges are blurred. Because the focus of this paper is on the normal direction, we did not go into the seed-search issue in detail. One simple way of finding seeds is to utilize the Hounsfield values of pixels, for which the value is defined to be 0 for water and either -1000 or -1023 for air. Muscle and bone are the principal tissues with Hounsfield values greater than that of water in CT slices. To find seeds near the edge of bone we first "windowed" the intensity, namely, we set those pixels with values less than zero to be zero, and set those pixels with values greater than some value c (say, 2000) to be c if in practice the intensity of bone is rarely greater than c. Now the edge strength of muscle and nonbone objects such as fat and air, and the edge strength of bone and such possible implanted materials as metals are much weakened, while the edge strength of muscle and bone keeps unchanged. Next, we calculated edge strength, h, for each pixel with intensity value within the "window." It was expected that the edge strength of muscle and bone was dominant but in practice it was found that the weakened edge strength of muscle and fat or air could still be very strong. To further filter out those nonbone edge seeds, we used the *weighted edge strength*

$$h \exp\left(-\frac{(I - \langle I \rangle)^2}{2\sigma_1^2}\right) \qquad (20)$$

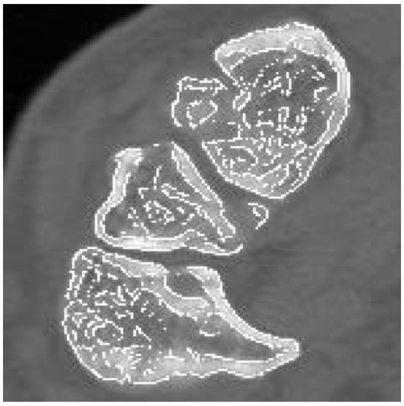

Fig. 7. Detected initial seeds (white points) by (20) in a CT slice of a human pelvis.

where I is the intensity of pixel within the "window," $\langle I \rangle$ and σ_1 are, respectively, the expected value and variance of I. This equation favors pixels with intensity values close to the expected value, and filters pixels corresponding to tissues other than bone (e.g., muscle/fat edges or edges due to metallic implants).

Finally, we sorted these seeds according to their weighted edge strength from the largest to the smallest and used only those seeds with large weighted edge strengths. An example is shown in Fig. 7, where a CT slice of a human pelvis has the seed points superimposed in white. Most pixels at the edges of bones in the image were usable as seeds, but there were also many seeds interior to the bones and a few seeds between the bones. The seeds within the bones generate edges that were removed in a subsequent contour-analysis process. However, the seeds between the bones produced edges that connected distinct bones, which were deleted by postprocessing.

G. Deletion of Edges Between Bones

In order to aid in segmentation, which is the identification of distinct bones or bone fragments in a given CT slice, we postprocess the edge map to eliminate edges that join contours of distinct bones, which we call *inter-osseal* edges. These are troublesome to automatically delete because they often occur in regions where edge strength is weak. Simple thresholding and hysteresis techniques do not appear to be promising for deleting inter-osseal edges.

Our algorithm often detects inter-osseal false edges when two bones are either nearby and or have an indistinct region separating them (often only a few pixels wide). An example is shown in Fig. 8(a). We refer to a point at which two or more edges meet as a *knot*, and we classify a knot and its neighboring detected edge points into three groups I, II, and III based on their normal directions. Because the normal direction of edges in nearby but distinct bones must be very different (almost anti-parallel), we put pixels with similar normal directions to that of a knot into Group I; this grouping can be generous so, for example, we require that the difference of the normal direction of a Group I edge and the knot be less than 45°. Those edges with normal directions very different from that of the knot, for example, greater than 135°, are classified into Group II. All other edges are classified into Group III.

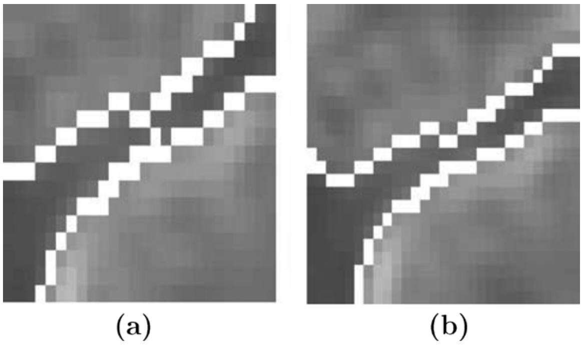

Fig. 8. False edge deletion by using normal direction. (a) Before the false edge connecting bones is deleted. (b) After the false edge is deleted.

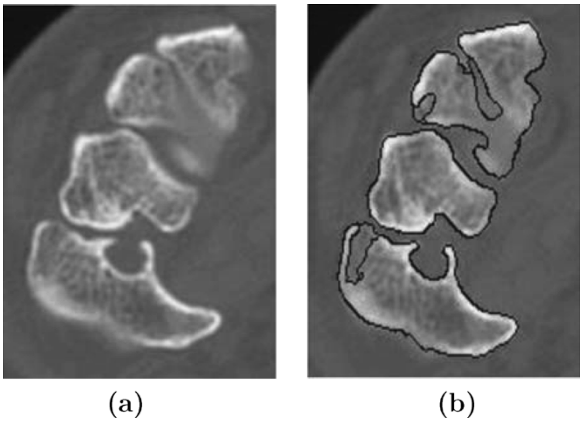

Fig. 9. (a) One blurred CT slice of a human wrist. (b) The edge map of the slice obtained by our algorithm.

For each edge in group I, we search for the nearest neighbor in group II. A line passing through these two pixels can be constructed. If a third edge belongs to Group III and is on the line between the first two edges, we classify it as an inter-osseal edge point and it is deleted. Fig. 8(b) shows the result after the inter-osseal edges have been deleted.

When a given CT slice is too noisy or ill-structured, some false edges may still exist, as is illustrated in Fig. 9. In such cases our edge-tracing approach based on normal direction cannot segment all the contours of bone correctly, and manual processing of the edge map is required.

III. Experimental Results

We have conducted extensive experiments of our algorithm for segmenting human wrist, leg, and pelvis CT slices from patients who volunteered in a study approved by the Research Ethics Board of Queen's University and Kingston General Hospital. The data sets were acquired using a commercial CT scanner (General Electric, Milwaukee, WI) with 12-bit pixels, 512 × 512 per slice. The typical thickness and spacing of slices were 2 mm for leg and pelvis scans, and 1 mm for wrist scans. Pixel sizes, in millimeters, were typically: in leg scans, 0.52 × 0.52; in pelvis scans, 0.66 × 0.66; and in wrist scans, 0.23 × 0.23. These patients were scheduled for orthopedic surgery and exhibited a wide range of bone pathology and morphometric variation, which increased the difficulty of automatic

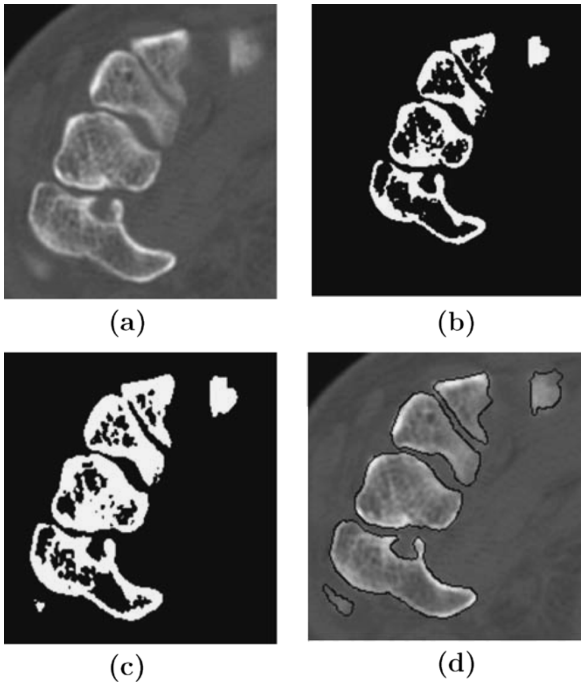

Fig. 10. Comparison of the segmentation by global thresholding and our algorithm. (a) The original CT slice. (b) The segmentation using a global threshold of 360. (c) The same as (b), using a global threshold of 250. (d) The edge map computed with our algorithm.

segmentation. Specific challenges were that, in the wrist, the carpal bones are very close to each other; in the knee, the femur and tibia overlap in some horizontal CT slices because of the diarthroidal joint; and in the pelvis, the various bones are of highly variable density both physically and radiologically. Although these anatomical regions are very different, in our tests we used the fixed parameter values of $\sigma_{\min} = 0.1$, $\sigma_{\max} = 1.2$, $\Delta\theta_{\text{tol}} = 10^{\circ}$ to test the robustness of our algorithm. On a Sun station (UltraSPARC-III, 750-MHz processor, 1 GB RAM memory) our algorithm took 2 to 3 s per slice to complete the edge-finding process, including postprocessing. We compared the performance of our algorithm with some other segmentation methods on a representative set of images.

A. A Comparison with Global-Thresholding Segmentation

Global thresholding is perhaps the most commonly used method in the automated CT segmentation of bone because the intensity values of bone, especially for hard bone, are statistically higher than the intensity of the other tissues so there is an intensity threshold such that the intensities of most pixels of bone are greater than the threshold value. However, since bone is in fact a highly inhomogeneous tissue, it is seldom possible to find a threshold value that is less than values of all bone *and* greater than values of all other tissues. Fig. 10(b) shows the segmentation result on the image in Fig. 10(a) when the threshold was set to 360 Hounsfield units. There are many discontinuities of the contours of the bones, and the cancellous bone at the bottom left is missing. To close the contours, the threshold should be lower, but a lower threshold causes false connections, as is evident in Fig. 10(c) when the threshold was reduced to 250 Hounsfield units. There are still some unclosed contours, but false connections have appeared. In contrast, the segmentation result from our algorithm is displayed in Fig. 10(d). All the contours of the bones in the slice were detected and closed, and there were no false connections.

B. Comparison with Snake-Based Segmentation

The snakes method is a very popular approach in medical image segmentation. The algorithm was originally proposed by Kass *et al.* [2], and many authors have contributed variations to improve the performance [10], [37], [38]. In this section, we use the greedy algorithm for snakes [10] to check its performance.

The snakes method requires an initial estimate of the contour. To fully automate the method, the initial contour should be provided with little or no human intervention. One possible approach is to use the segmented result of the neighboring slice as the initial contour in the current slice; another approach is to use an anatomical atlas to provide an initial contour. A well-known characteristic of snakes is that their performance is sensitive to contour initialization, and the initial contour provided by the above approaches may not be sufficiently accurate to drive the snakes algorithm to a correct result. In the following experiment, we manually selected some initial points that were close to the edge of the object of interest, and then linearly connected these points to form a closed initial contour. This initial contour was a close but not exact approximation of the object contour. The parameters, i.e., the weights of the energy terms, were adjusted to improve the segmented results.

We again used the slice in Fig. 10(a) to examine the performance of the snakes method. This slice is both challenging and of particular interest because it has both cancellous bone and bones that are very close, which often occurs in CT slices. Fig. 11(a) and (b) shows the initial and final contours of the bone. It can be seen that the final contour diverges from the bone even though the initial points, especially those between the two bones, were chosen to be very close to the true edge of the bone. In Fig. 11(c) and (d), when the initial contour was not close to the concave edge of the bone, the algorithm failed to detect the concave bone edge. Finally, when the algorithm was used to segment cancellous bone, the total energy did not reach the minimum but increased as shown in Fig. 11(f). In this case, the final contour shown in Fig. 11(e) was not reliable.

We found that, even for a good initialization, it was difficult for snakes to find the true edge of a bone that was concave, cancellous, or that was very close to another bone. By comparing Fig. 11(b), (d), (e) with Fig. 10(d), it can be seen that our algorithm was more successful at detecting the bone contours in this image, which is a representative of our human CT slices.

Recently, multiple active contour models have extended snakes into 3-D so that more information can be used to help control the sensitivity of the snakes on the initial contour and, therefore, improve the performance [39]. Our current scheme based on normal direction can also be extended to 3-D, for which we expect that the performance will be improved because the accuracy of estimated normal directions can be improved using the additional information.

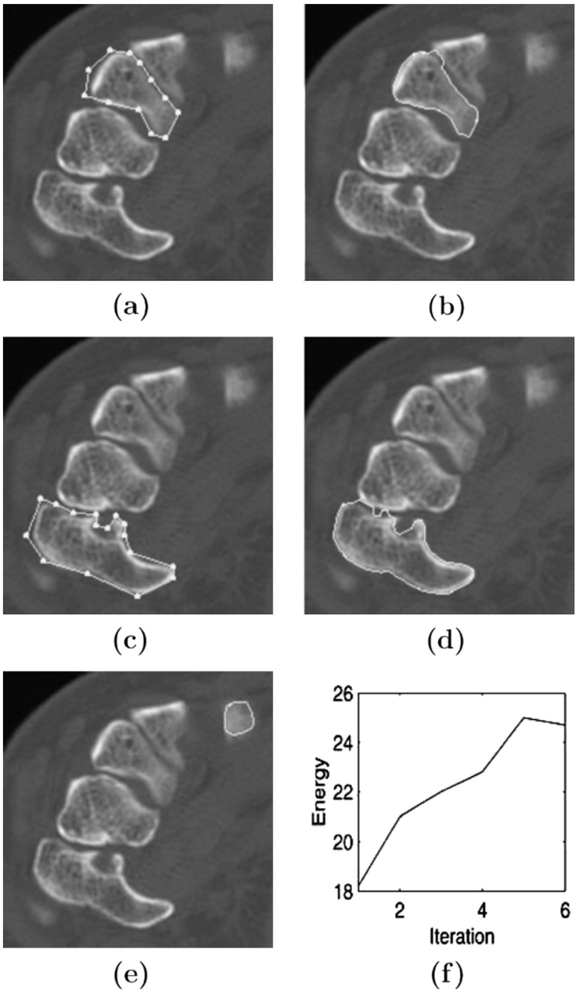

Fig. 11. Examples to demonstrate the performance of snakes on a wrist CT slice. (a), (c) The initial contours and (b), (d) the corresponding final contours. (e) The final contour and (f) the total energy versus iteration time.

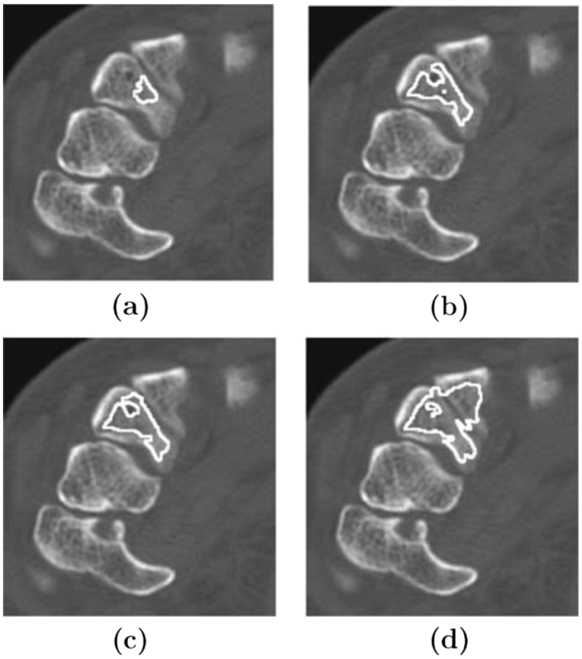

Fig. 12. Examples to demonstrate the performance of Geodesic active contour level sets on a wrist CT slice. The superimposed white curves stand for the stable zero level set when σ is (a) 1.6, (b) 2.0, (c) 2.4, and (d) 2.5. It can be seen that when $\sigma \leq 2.4$, the zero level set can not reach the edge of the bone, and when $\sigma = 2.5$, the zero level set goes to another bone.

C. A Comparison With Level Set-Based Segmentation

A level set method solves a partial differential equation, typically

$$\frac{d\psi}{dt} = -\alpha \boldsymbol{A}(\boldsymbol{x}) \cdot \triangledown\psi - \beta P(\boldsymbol{x})|\triangledown\psi| + \gamma Z(\boldsymbol{x})\kappa|\triangledown\psi| \quad (21)$$

where $\mathbf{A}$, $\mathbf{P}$ are the advection and propagation terms, respectively, and Z is the spatial modifier for the mean curvature κ. The scalar α, β, γ weight the relative influence of each of the terms on the movement of the interface.

We used level set procedures in ITK [40] to segment bones since ITK has become a standard software for medical image processing and it has implemented many level set algorithms. Except *CannySegmentationLevelSetImageFilter* and *LaplacianSegmentationLevelSetImageFilter* which need initial models, we tested all the other level set procedures and found that the procedure *GeodesicActiveContourImageFilter* performed best in segmenting bones. This procedure needs the user give seven parameter values including α, β, the variance σ, and the position of seeds (used by *FastMarchingImageFilter* to generate an initial contour for *GeodesicActiveContourImageFilter*). We focused on the performance under various σ values. We tested the procedure on the wrist bone in Fig. 10(a) using many parameter sets for each σ and chose the best, which is displayed in Fig. 12. The performance of the procedure is sensitive to σ, and does not appear to be suitable for segmenting bones. However, we note that the procedure was not particularly designed for segmenting bones. To improve the performance of level sets in segmenting bones, bone properties should be considered in the level set description (21).

D. Segmentation With and Without Normal-Direction Correction

One of the important steps in our algorithm is the correction of the estimated normal direction. For hard bone, the reliability of the estimated normal direction is close to 1; in such cases our correction scheme will have little effect on the segmentation results. For cancellous bone and extremely close bones, however, the correction is very useful. Fig. 13 displays such a comparison on three consecutive CT slices of human pelvis. Fig. 13(a), (d), (g) shows the original slices. There are roughly three bones in each slice: the left and right ilia, and the sacrum. The ilia are hard, weight-bearing bones but in this image some of their regions appear to be blurred.[2] The edge of the sacrum is weak, and some parts of the bone are particularly difficult to see. The Hounsfield values of the pixels at these parts are almost the same as that of muscle. Further, the sacrum is close to the ilia, and their boundaries are blurred. Roughly speaking, among them, slice (a) is the clearest and slice (g) is the most blurred.

Fig. 13(b), (e), (h) shows the segmentation results on these slices, respectively, without the normal-direction correction step, namely, by setting the reliability equal to 1. It can be seen

[2] This blurring is likely from volume averaging during CT acquisition.

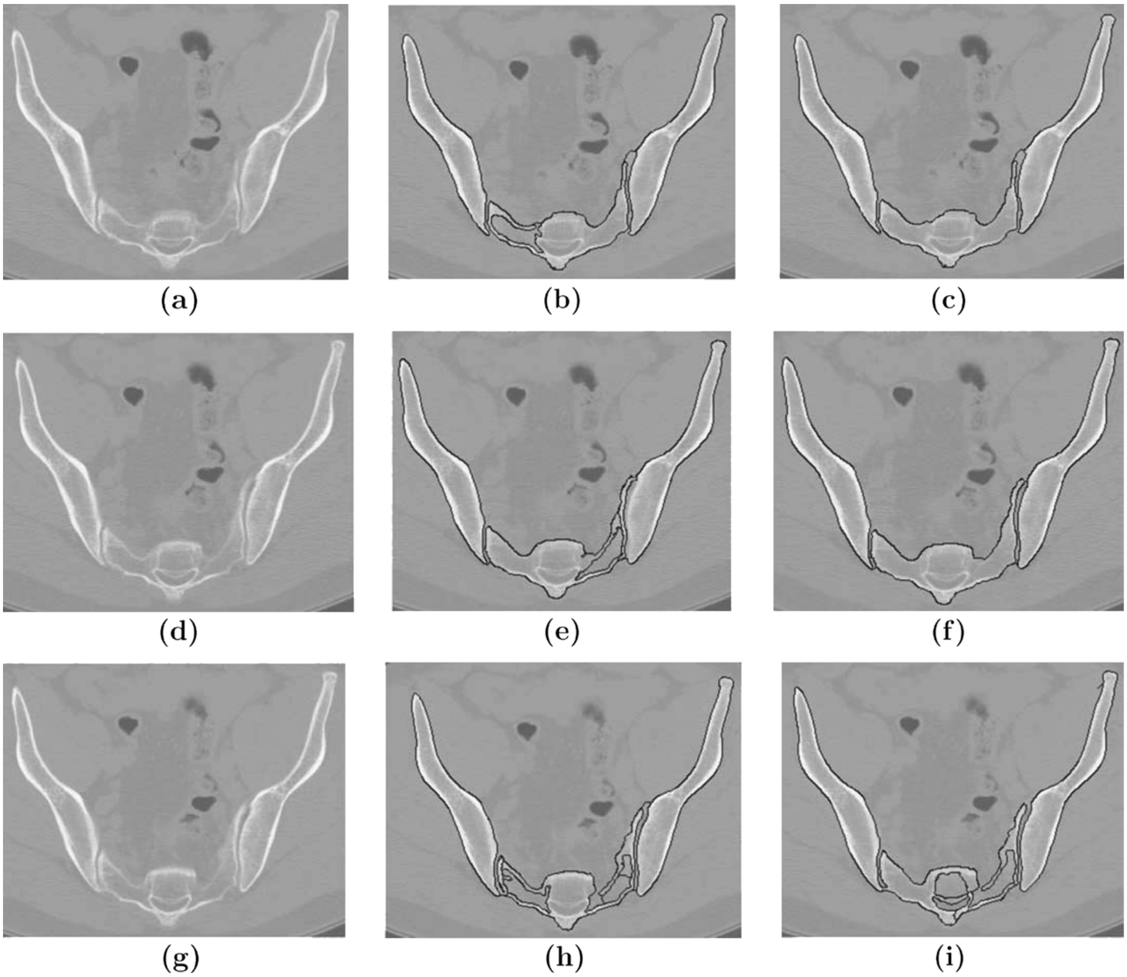

Fig. 13. Comparison of segmentation with and without normal-direction correction. (a), (d), (g) Three contiguous original CT slices of a human pelvis. (b), (e), (h) The edge maps, respectively, of the images without correcting the normal direction. (c), (f), (i) The edge maps with the normal direction corrected.

that edge map (b) was segmented well, except for an inter-osseal edge between the sacrum and the right ilium. In edge map (e), four inter-osseal edges appeared between the sacrum and ilia, and one contour of the right ilium was discontinuous; this break caused the entire detected but not closed contour to be lost after the postprocessing which floods the closed contours only. In edge map (h), again there were three inter-osseal edges between the sacrum and ilia, and the contour of the sacrum was interrupted at two places.

If the normal direction was corrected, the segmentation results were much improved as shown in Fig. 13(c), (f), (i), which were segmented from slices (a), (d), and (g), respectively. Edge maps (c) and (f) were segmented well, except for some short inter-osseal connections. For edge map (i), there was one inter-osseal edge between the sacrum and the left ilium, and one discontinuity in the contour of the sacrum. Edge map (i) is more accurate than edge map (h).

Our algorithm has been tested on more than 1000 CT slices of the human wrist, leg (including knee) and pelvis *with the same set of parameters*. Unfortunately, there is no commonly accepted set of CT slices for benchmarking the performance, nor are there yet any agreed-upon rules by which to compare various algorithms. In our case, the two drawbacks in the edge map are inter-osseal edges and contour discontinuities. Inter-osseal edges are those edges that need to be removed prior to final flooding, which is the process of setting the intensity of a closed region to a uniform value. For example, in Fig. 14, a section of Fig. 13(h) is shown to contain three inter-osseal edges between the sacrum and ilia, highlighted by circles 1, 3, and 4, and there are two discontinuities in circles 1 and 2. False edge 1 between the sacrum and the left ilium is not part of a closed contour, and it would disappear after flooding. In contrast, false

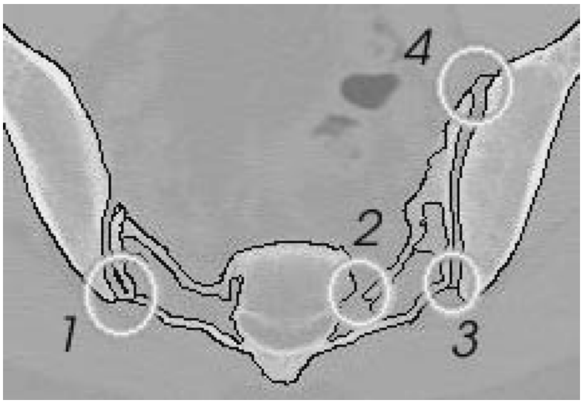

Fig. 14. Example to show inter-osseal edges and contour discontinuities.

TABLE I
PERFORMANCE OF OUR ALGORITHM IN PELVIS, LEG, AND WRIST CT SLICES WITH (WITHOUT) NORMAL-DIRECTION CORRECTION

CT Slices	n	n_i	n_d	Performance(%)
Pelvis	219	10(18)	33(85)	96.6(91.4)
Leg	230	10(47)	58(83)	94.5(90.7)
Wrist	46	5(5)	8(9)	95.5(95.0)

edges 3 and 4 between the sacrum and the right ilium do form a closed contour with the right side of the sacrum and the left side of the right ilium, so one of these false edges must be broken so that the contour is not closed. There are also some unexpected edges in the sacrum. These do not have to be specifically accommodated because they will disappear by flooding after the outer contour of the sacrum is closed. Thus, for Fig. 14, only one false edge needed to be broken.

In a clinical scenario, the main purpose of an automated segmentation algorithm is to reduce the time of manual intervention by a skilled professional, although certainly other aspects such as accuracy and efficiency should also be considered [41]. To show the performance of our algorithm from this viewpoint, based on our experience, we assume that on average a skilled operator needs 10 s to manually segment a slice, 1 s to break a false edge and 2 s to repair a discontinuity, which are typically small (< 6 pixels). Then the performance of our algorithm may be estimated as

$$\mathrm{Performance} = 1 - \frac{(\mathrm{n_i} + 2\mathrm{n_d})}{(10\mathrm{n})} \tag{22}$$

where n_i is the number of inter-osseal edges and n_d is the number of contour discontinuities in the total number (n) of edge maps (the number of slices). We have used (22) to come up with a quantitative measure to evaluate the performance of our algorithm.

Table I shows the performance of our algorithm with and without the normal-direction correction scheme. When the normal direction correction scheme was used, in 219 edge maps of the pelvis CT scan there were 10 inter-osseal edges that needed to be broken before flooding, and 33 discontinuities, so the performance is 96.6%. For the full-leg CT scan which had 230 slices in total, there were 10 inter-osseal edges and 58 discontinuities, for a performance of 94.5%. In this particular patient's leg, there was also a metal implant that appeared in 85 of the 230 slices. Metal typically causes serious reconstruction artifacts of the images [42], which are clearly visible in Fig. 15(a) and (c). When the metal artifacts dominated the image, the performance of our algorithm degraded [see Fig. 15(b)]. Interestingly, when the bone was also hard, our algorithm produced satisfactory results as shown in Fig. 15(d). If we consider only the 145 slices without metal artifacts, then there were 6 inter-osseal edges and 15 discontinuities, so the performance rating was 97.5%. For those slices with iron, the performance was 89.9%.[3] Finally, for the total of 46 wrist slices, there were 5 inter-osseal edges and 8 discontinuities, so the performance was 95.5%.

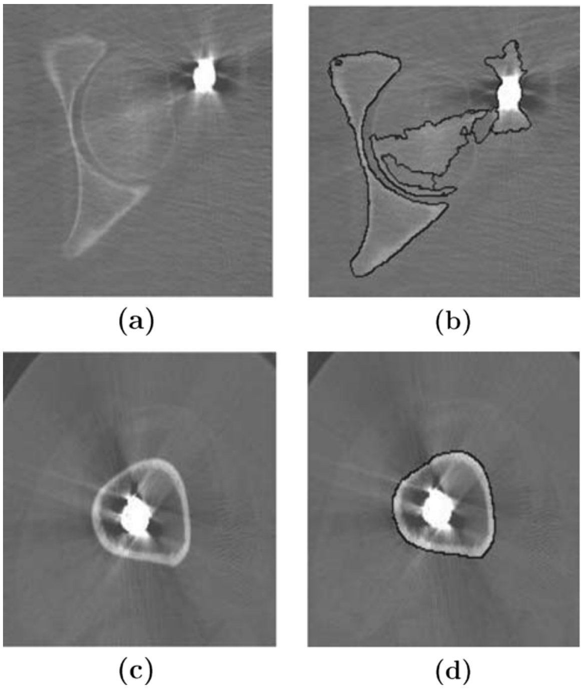

Fig. 15. Segmentation of CT slices with iron by our algorithm. (a), (c) The original CT slices of a human leg. (b), (d) The edge maps of slices (a) and (c), respectively.

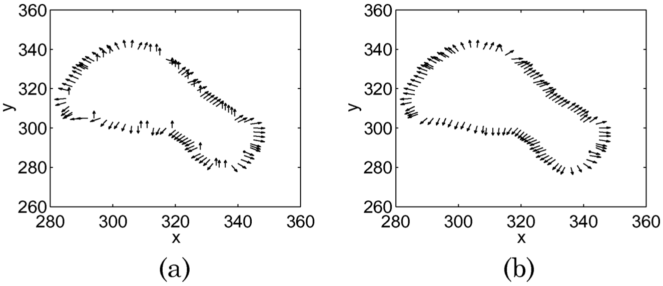

Fig. 16. Estimated normal direction of the edge of a bone by the single- scale approach (a) with $\sigma = 1.2$, and by the multiscale approach (b) with $\sigma_{\min} = 0.1, \sigma_{\max} = 1.2$.

Without normal-direction correction, the performance of our algorithm on the pelvis and leg degraded by about 5%. For those slices with metal artifacts, there were a total of 19 inter-osseal edges and 61 contour discontinuities, so the performance was 83.4%. For the wrist, the performance was almost the same as that with the correction scheme. This was because, except for some very blurred slices which could not be segmented well by either procedure, the edge strength of bones in the wrist CT slices allowed the reliability measure to be set to 1.

E. Segmentation With Single-Scale and Multiscale

In Section II, we argued that the normal direction estimated by the single-scale approach is not reliable and that our multiscale approach described in Section II-D produces a more reliable normal direction. One result is shown in Fig. 16 which depicts the estimated normal directions at each pixel along the contour of one bone [shown in Fig. 10(a)]. Fig. 16(a) shows that

[3] We believe that it would be difficult for a skilled person to manually segment the slices with metal artifacts in just 10 s per slice.

TABLE II
PERFORMANCE OF OUR ALGORITHM WITH THE SINGLE-SCALE APPROACH AND WITH DIFFERENT σ

σ	n_i	n_d	Performance(%)
0.5	15	42	78.5
0.6	11	20	88.9
0.7	12	23	87.4
0.8	14	25	86.1
0.9	12	18	89.6
1.0	10	11	93.0
1.1	6	13	93.0
1.2	10	15	91.4
1.3	10	18	90.0
1.4	13	22	87.6
1.5	15	25	85.9
1.6	14	20	88.3
1.7	20	25	84.8
1.8	18	27	84.4
1.9	22	37	79.1
2.0	73	100	40.7

the normal direction estimated by the single-scale approach is completely wrong at many pixels: the normal direction should be *outward*, but at many pixels the direction was either *inward* or close to the tangent direction of the edge. The normal directions estimated by our multiscale approach, however, were all outward as shown in Fig. 16(b).

Estimation errors of the normal direction causes a lower performance of our algorithm. Table II lists the performance of our algorithm on 46 CT slices of the wrist when the single-scale approach is employed. A single value of $\sigma \in [0.5, 2.0]$ was used to smooth the image, to estimate the normal direction and to detect the edge. It was observed that the performance significantly changed with the value of σ, and that the best performance was 93.0% for $\sigma = 1.0$ and $\sigma = 1.1$. The performance for our multiscale approach was 95.5%, so the multiscale approach was better than the single-scale approach in our algorithm. In the tested slices, the bones in the wrist slices had the highest edge strength. The performance of the single-scale approach in the pelvis and leg CT slices was even worse than its performance on the wrist CT slices.

F. Comparison With Manual Segmentation

Accuracy of a segmentation is often evaluated by comparing the segmentation results with segmentations by experts [12], [41], which are used as the standard. Accuracy may be calculated as

$$\text{Accurancy} = 1 - \frac{\mathrm{n}_{diff}}{\mathrm{n}_{std}} \tag{23}$$

where n_{diff} is the number of pixels missed or over segmented by our algorithm. In other words, if pixel $p(x, y)$ is in a bone in the standard image, but is falsely detected not to lie in a bone by our algorithm, then we add one (1) to n_{diff}; similarly, if a nonbone pixel is falsely segmented as lying in a bone by our algorithm, then we also add one (1) to n_{diff}. For n_{std}, there are two considerations: when n_{std} is the number of pixels of the standard object, we are comparing the accuracy of area segmentation, and when n_{std} is the number of edge pixels of the standard object, (23) is for the accuracy of edge segmentation.

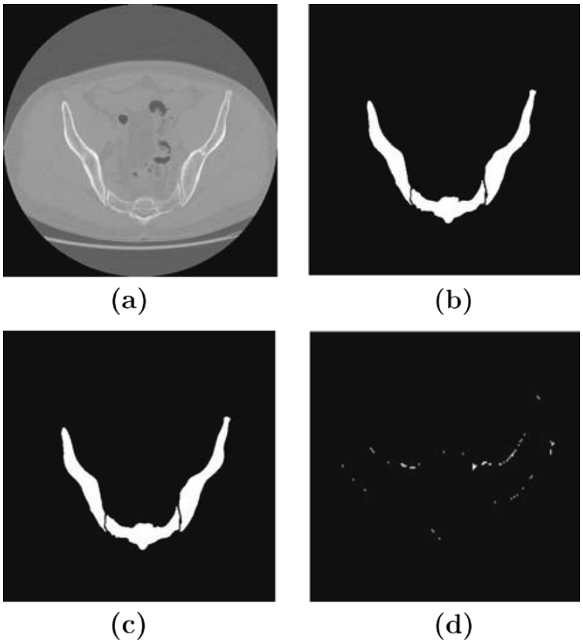

Fig. 17. Comparison of segmentations where (a) the original image, (b) the manually segmented image, (c) the segmented image by our algorithm, and (d) the difference of the segmentations.

We examined 9 CT slices of a human pelvis, which are difficult to segment even in manual way. Three of them are shown in Fig. 13(a), (d), and (g); the others were adjacent CT slices. An expert segmented the slices manually, and the result was taken as the standard. The slices were then automatically segmented using our algorithm, and the same expert manually corrected the inter-osseal connections and contour discontinuities. One example is shown in Fig. 17. It can be seen from Fig. 17(d) that the difference of these two segmentation results was mainly in the blurred region of the sacrum. For these 9 slices, the mean accuracy calculated with (23) was 98.9%, 95.1% for, respectively, the area and edge segmentation comparisons, and the standard deviations are 0.391% and 0.502%, respectively. Based on this comparison and the consideration that these 9 slices are among the most difficult for segmentation, we expect that our algorithm be as accurate in segmenting similar cases.

IV. CONCLUSION

Accurately estimated normal direction of an edge is of general utility in image processing. Estimation using a single scale (the value of σ in the first derivative of the Gaussian) may cause large errors, especially when the estimated edge is between two bones close to each other. We have proposed an approach to estimate the normal direction in multiple scales. The scales are automatically determined by assuming that the normal direction of an edge does not change significantly with scale. However, even such an estimated normal direction is not completely reliable because of image quality. To segment cancellous bones, where the edge strength is weak and image quality is low, we have further corrected the estimated normal direction by assuming that contours of bone are smooth and continuous. We have shown that our correction scheme can be expressed as a 1-D Kalman filter. In addition, the Shen-Castan optimal linear filter for mono-step edges can be taken as a special case of this Kalman filter, and the filter in our correction scheme approximates the distribution of the estimated normal directions along the contour (which is why our correction scheme works well). In calculating the weight in the scheme, unlike in the Kalman filter which uses the measurement and prediction variances, we have measured the *reliability* of the estimated normal direction and have determined the reliability during estimation of the normal direction. Furthermore, we have obtained an optimal scale σ *for each edge point* during this normal-direction estimation. The value of σ was used in the first derivative of a Gaussian filter, which was the edge detector in our algorithm. Because the normal direction was corrected and the value of σ was optimized, our algorithm is robust. We can use a global set of parameters to segment different CT slices, even those with significant artifacts that arise from metal implants in patients.

We also used the corrected normal direction in a postprocessing step to delete false connections between nearby bones. False edges are very difficult to distinguish from real edges because they often appear in the blurred region of the image. The normal direction, which encodes geometrical information of the edge, can be used to classify the neighboring edge points into three groups. Most of these false inter-osseal edges were removed by our technique.

In order to automatically segment bone, we developed an approach to find initial seeds in two steps. First, pixels of bone and muscle were found based on their Hounsfield values greater than that of water. Second, the edge strength of these pixels were calculated, and the initial seeds were then determined from the modified edge-strength histogram. By this preprocessing, the contours of nonbone tissues seldom appear in the edge map, which in turn increases the speed of the segmentation.

Our algorithm has been tested on CT slices of a pelvis, a leg (including the knee joint), and wrist. We compared our algorithm to other standard methods, measuring performance based on the number of inter-osseal edges and contour discontinuities. Our experiments showed that the correction of the normal direction improved the segmentation of cancellous bone. The experimental results also showed that our algorithm was fast ($<$ 3 s per slice), and the accuracy is very high when compared to manual segmentation by a human expert. In the near future, we will employ this algorithm in our image-guided surgical suite and evaluate its performance on patients.

In this work, we focused on 2-D CT images to show the advantage of accurately computing the normal direction of an edge. We plan to extend our algorithm into 3-D, where information in neighboring slices can be utilized to improve the accuracy of the normal-direction estimation. Although multi-scale analysis of CT images will not change in 3-D, some quantities in our scheme such as the re-scale factor γ of the reliability may become more stable from slice to slice when we use 3-D information to estimate the normal direction of bone edges. Thus, it is expected that the 3-D extension will shorten the segmentation time and improve the performance.

ACKNOWLEDGMENT

The authors gratefully acknowledge Dr. Y. Chen, S. Leclaire, and I. Wang for fruitful discussions on this work, and N. Chen for offering her code of the snakes algorithm.

REFERENCES

[1] M. L. Comer and E. J. Delp, “Segmentation of textured images using a multiresolution gaussian autoregressive model,” *IEEE Trans. Image Process.*, vol. 8, no. 3, pp. 408–420, Mar. 1999.

[2] M. Kass, A. Witkin, and D. Terzopoulus, “Snake: active contour models,” *Int. J. Comput. Vis.*, vol. 1, no. 4, pp. 321–331, 1988.

[3] D. Wang, “A multiscale gradient algorithm for image segmentation using watersheds,” *Pattern Recognit.*, vol. 30, no. 12, pp. 2043–2052, 1997.

[4] M. Garza-Jinich, P. Meer, and V. Medina, “Robust retrieval of three-dimensional structures from image stacks,” *Med. Image Anal.*, vol. 3, no. 1, pp. 21–35, 1999.

[5] R. Adams and L. Bischof, “Seeded region growing,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 16, no. 6, pp. 641–647, Jun. 1994.

[6] H. Tek and B. B. Kimia, “Volumetric segmentation of medical images by three-dimensional bubbles,” *Comput. Vis. Image Understand.*, vol. 64, no. 9, pp. 884–900, 1997.

[7] R. Kohler, “A segmentation system based on thresholding,” *Comput. Graph. Image Process.*, vol. 15, pp. 319–338, 1981.

[8] J. Bernsen, “Dynamic thresholding of grey-level images,” presented at the *8th Int. Conf. Pattern Recognition*, Paris, France, Oct. 1986.

[9] N. Otsu, “A threshold selection method from gray-level histograms,” *IEEE Trans. Syst., Man, Cybern,*, vol. SMC-9, no. 1, pp. 62–66, 1979.

[10] D. Williams and S. Mubarak, “A fast algorithm for active contours and curvature estimation,” *CVGI: Image Understand.*, vol. 55, no. 1, pp. 14–26, 1992.

[11] M.-H. R. Cardinal, E. Thérasse, J. Meunier, G. Soulez, and G. Cloutier, “Intravascular ultrasound image segmentation: A fast-marching method,” in *Lecture Notes in Computer Science*, R. E. Ellis and T. M. Peters, Eds. Berlin, Germany: Springer-Verlag, 2003, Proc. MICCAI’03, pp. 432–439.

[12] J.-P. Guyon, M. Foskey, J. Kim, Z. Firat, B. Davis, K. Haneke, and S. R. Aylward, “VETOT, Volume estimation and tracking over time: Framework and validation,” in *Lecture Notes in Computer Science*, R. E. Ellis and T. M. Peters, Eds. Berlin, Germany, 2003, Proc. MICCAI’03, pp. 142–149.

[13] P.-F. D’Haese, V. Duay, T. E. Merchant, B. Macq, and B. M. Dawant, “Atlas-based segmentation of the brain for 3-dimensional treatment planning in children with infratentorial ependymoma,” in *Lecture Notes in Computer Science*, R. E. Ellis and T. M. Peters, Eds. Berlin, Germany: Springer-Verlag, 2003, Proc. MICCAI’03, pp. 627–634.

[14] J. A. Sethian, *Level Set Methods and Fast Marching Methods*. Cambridge, U.K.: Cambridge Univ. Press, 1999.

[15] H. K. Zhao, T. Chan, B. Merriman, and S. Osher, “A variational level set approach to multiphase motion,” *J. Comput. Phys.*, vol. 127, pp. 179–195, 1996.

[16] V. Caselles, R. Kimmel, and G. Sapiro, “Geodesic active contour,” *Int. J. Comput. Vis.*, vol. 22, no. 1, pp. 61–97, 1997.

[17] J. R. Beveridge, J. Griffith, R. R. Kohler, A. R. Hanson, and E. M. Riseman, “Segmenting images using localized histograms and region merging,” *Int. J. Comput. Vis.*, vol. 2, no. 3, pp. 311–352, 1989.

[18] S. C. Zhu and A. L. Yuille, “Region competition: unifying snakes, region growing, and bayes/mdl for multiband image segmentation,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 18, no. 9, pp. 884–900, Sep. 1996.

[19] L. Vincent and P. Soille, “Watersheds in digital spaces: an efficient algorithm based on immersion simulations,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 13, no. 6, pp. 583–598, Jun. 1991.

[20] T. B. Sebastian, H. Tek, J. J. Crisco, and B. B. Kimia, “Segmentation of carpal bones from ct images using skeletally coupled deformable models,” *Med. Image Anal.*, vol. 7, no. 1, pp. 21–45, 2003.

[21] H. Soltanian-Zadeh and J. P. Windham, “A multiresolution approach for contour extraction from brain images,” *Med. Phys.*, vol. 24, no. 12, pp. 1844–1853, 1997.

[22] J. Canny, “A computational approach to edge detection,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 8, no. 6, pp. 679–714, Jun. 1986.

[23] D. Mumford and J. Shah, “Optimal approximation by piecewise smooth functions and associated variational problems,” *Comm. Pure Appl. Math.*, vol. 42, pp. 577–685, 1989.

[24] D. P. Dobkin, A. R. Wilks, S. V. F. Levy, and W. P. Thurston, “Contour tracing by piecewise linear approximations,” *ACM Trans. Graphics*, vol. 9, no. 4, pp. 389–423, 1990.

[25] S. Sarkar and K. L. Boyer, “Optimal infinite impulse response zero crossing based edge detectors,” *Comput. Vis., Graph., Image Process.*, vol. 54, no. 2, pp. 224–243, 1991.

[26] J. Shen and S. Castan, “An optimal linear operator for step edge detection,” *Comput. Vis., Graph., Image Process.*, vol. 54, no. 2, pp. 112–133, 1992.

[27] T. Poggio, H. Voorhees, and A. Yuille, “A Regularized Solution to Edge Detection,” MIT Artif. Intell. Lab., 1985.

[28] W. Yao and R. E. Ellis, A piecewise-linear map for finding edges in images, in Dynamics of Continuous, Discrete and Impulsive Systems, 2003.

[29] A. P. Paplinski, “Directional filtering in edge detection,” *IEEE Trans. Image Process.*, vol. 7, no. 4, pp. 611–615 , Apr. 1998.

[30] V. Torre and T. Poggio, “On edge detection,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-8, no. 2, pp. 147–163, Feb. 1986.

[31] R. E. Kalman, “A new approach to linear filtering and prediction problems,” *Trans. ASME J. Basic Eng.*, vol. 82D, pp. 35–45, 1960.

[32] R. G. Brown and P. Y. C. Hwang, *Introduction to Random Signals and Applied Kalman Filter*, 2nd ed. New York: Wiley, 1992.

[33] G. Welch and G. Bishop, “An introduction to the Kalman filter,” Tech. Rep., Dept. Comput. Sci., Univ. North Carolina, Chapel Hill, 2002.

[34] A. L. Yuille and T. A. Poggio, “Scaling theorems for zero crossings,” *IEEE Trans. Pattern Anal Mach. Intell.*, vol. PAMI-8, no. 1, pp. 12–25, Jan. 1986.

[35] J. Babaud, A. P. Witkin, M. Baudin, and R. O. Duda, “Uniqueness of the gaussian kernel for scale-space filtering,” *IEEE Trans. Pattern Anal Mach. Intell.*, vol. PAMI-8, no. 1, pp. 26–33, Jan. 1986.

[36] T. Lindeberg, “Scale-space for discrete signals,” *IEEE Trans. Pattern Anal Mach. Intell.*, vol. PAMI-12, no. 3, pp. 234–254, Mar. 1990.

[37] A. A. Amini, S. Tehrani, and T. E. Weymouth, “Using dynamic programming for minimizing the energy of active contours in the presence of hard constraints,” in *Proc. Int. Conf. Comput. Vis.*, 1988, pp. 95–99.

[38] L. L. Ji and H. Yan, “Attractable snakes based on the greedy algorithm for concour extraction,” *Pattern Recognit.*, vol. 35, pp. 791–806, 2002.

[39] V. Chalana, D. T. Linker, D. R. Haynor, and Y. Kim, “A multiple active contour model for cardiac boundary detection on echocardiographic sequences,” *IEEE Trans. Med. Imag.*, vol. 15, no. 3, pp. 290–298, Jun. 1996.

[40] (2002) Insight Segmentation and Registration Toolkit. The National Library of Medicine, DC. [Online]. Available: http://www.itk.org

[41] S. D. Olabarriaga and A. W. M. Smeulders, “Interaction in the segmentation of medical images: A survey,” *Med. Image Anal.*, vol. 5, pp. 127–142, 2001.

[42] J. A. Parker, *Image Reconstruction in Radiology*. Boca Raton, FL: CRC, 1990.