

NIH Public Access

Author Manuscript

Comput Med Imaging Graph. Author manuscript; available in PMC 2012 January 1

Published in final edited form as:

Comput Med Imaging Graph. 2011 January ; 35(1): 64-80. doi:10.1016/j.compmedimag.2010.09.007.

Improved Tensor Scale Computation with Application to Medical Image Interpolation

Ziyue Xu^{1,*}, Milan Sonka^{1,†}, and Punam K Saha^{1,2,‡}

¹ Department of Electrical and Computer Engineering, University of Iowa, Iowa City, IA 52242

² Department of Radiology, University of Iowa, Iowa City, IA 52242

Abstract

Tensor scale (t-scale) is a parametric representation of local structure morphology that simultaneously describes its orientation, shape and isotropic scale. At any image location, t-scale represents the largest ellipse (an ellipsoid in three dimensions) centered at that location and contained in the same homogeneous region. Here, we present an improved algorithm for t-scale computation and study its application to image interpolation. Specifically, the t-scale computation algorithm is improved by: (1) enhancing the accuracy of identifying local structure boundary and (2) combining both algebraic and geometric approaches in ellipse fitting. In the context of interpolation, a closed form solution is presented to determine the interpolation line at each image location in a grav level image using t-scale information of adjacent slices. At each location on an image slice, the method derives normal vector from its t-scale that yields trans-orientation of the local structure and points to the closest edge point. Normal vectors at the matching twodimensional locations on two adjacent slices are used to compute the interpolation line using a closed form equation. The method has been applied to BrainWeb data sets and to several other images from clinical applications and its accuracy and response to noise and other imagedegrading factors have been examined and compared with those of current state-of-the-art interpolation methods. Experimental results have established the superiority of the new t-scale based interpolation method as compared to existing interpolation algorithms. Also, a quantitative analysis based on the paired t-test of residual errors has ascertained that the improvements observed using the t-scale based interpolation are statistically significant.

Keywords

Tensor scale; local scale; interpolation line; medical image interpolation; medical imaging

1 INTRODUCTION

Scale [1–4] plays an important role in many medical imaging applications and is useful in determining the optimum trade-off between noise smoothing and perception/detection of

^{© 2010} Elsevier Ltd. All rights reserved.

Correspondence to: Punam K Saha.

^{*}ziyue-xu@uiowa.edu

[†]milan-sonka@uiowa.edu

[‡]punam-saha@uiowa.edu

Publisher's Disclaimer: This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting proof before it is published in its final citable form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

structures. It may be thought of as the spatial resolution, or, more generally, a range of resolutions needed to ensure a sufficient yet compact representation of target information [1]. Witkin [2] and Koenderink [3] mathematically formulated the concept of scale in the form of scale-space theory. Discrete scale-space representations [4] have been used in several imaging applications including segmentation [5], clustering [6], classification [7], and structural analysis [8]. The notion of "local scale" [9–12] emerged from the needs of spatially tuning neighborhood kernel size [13] or developing space-variant parameter controlling strategies [14] toward improving the performance of different methods. Local structure-based morphometric scale information would be useful in several applications including filtering, edge detection, object segmentation, registration, and analysis of regional structural properties. We refer to this notion of scale as "local morphometric scale" and briefly "local scale".

Saha et al. [13, 14] initiated the notion of local morphometric scale using a spherical model that was applied to image segmentation [13, 15, 16], filtering [14], registration [17], and removal of partial volume effects in rendering [18]. Although the preliminary results have demonstrated effectiveness of this notion of local scale in different image processing applications, a major limitation of the spherical model is that it ignores orientation and anisotropy of local structures. To overcome these limitations, Saha et al. [19, 20] introduced the notion of tensor scale (abbreviated as "t-scale") - a parametric representation of the largest ellipse (an ellipsoid in three dimensions) centered at the given image point and contained inside the same homogeneous region. Effectiveness of t-scale in image segmentation [21], registration [22], filtering [20] and also in quantifying local morphometry in complex quasi random networks of trabecular bone [23, 24] have been studied. Andalo et al. presented an efficient computational solution for t-scale and demonstrated its usefulness in detecting salient points on a given contour [25, 26]. Although the computational framework for t-scale as proposed by Saha et al. is robust and effective, optimization of individual steps has mostly been ignored. One major contribution of this paper is to improve the steps needed for t-scale computation. Specifically, we have optimized digital LoG and DoG kernels for edge detection at varying kernel sizes and edge characteristics, and have developed an effective ellipse fitting algorithm by combining both algebraic and geometric features.

Another major contribution of the current paper is the presentation of a new t-scale based image interpolation method that produces a closed form solution for interpolation of lines at individual image points in a gray level image. Medical images are commonly represented as stacks of slices and, often, slice thickness is larger than in-plane resolution leading to anisotropic voxels. However, isotropic data commonly facilitate most image analysis tasks and therefore, image interpolation has become a popular pre-processing step to restore voxel isotropy. Image interpolation techniques may be classified into two groups [27]: (1) imagebased and (2) object-based methods. Image-based interpolating approaches including nearest neighbor, linear, spline, and kernel-based interpolation methods [28] are primarily based on local image intensities in adjacent slices and require no structural correspondence between two slices. Although image-based techniques are computationally faster, such methods often suffer from several artifacts caused by nonlinear structural deformities in the slice direction. On the other hand, object-based interpolation techniques determine point-wise structural deformation in successive slices to capture structural alterations in the slice direction. The primary difference among algorithms within this group is essentially in the method of building structural correspondence between successive slices. Many methods are proposed in this category, for example, shape-based methods [29], registration-based methods [30] and morphology-based methods [31]. Here, we present a new deterministic method using tscale to determine the structural correspondence and to compute interpolation lines at different image locations. In Section 2, we present an improved t-scale computation

algorithm followed by the theory and algorithms for a new t-scale based image interpolation method. In Section 3, the experimental plans and methods are outlined. The results are presented and discussed in Section 4.

2 THEORY AND ALGORITHMS

In this section, we present an improved method for two-dimensional t-scale computation along with a new theory and algorithm for t-scale-based medical image interpolation. We will use "image point" to refer to a two- or three-dimensional grid point while a "point" will refer to any location in the Euclidean 2- or 3-D space. In 2-D, we will use "pixel" to denote the spatial extent represented by an image point while "voxel" will be used to serve the same purpose in 3-D.

2.1 T-scale Computation

At a given image point p in 2-D, its t-scale is the parametric representation of the largest ellipse centered at p and contained in the same homogeneous region. T-scale at p is computed by locating edge points visible from p along different directions which are then used to compute the t-scale ellipse at p. Basic steps of t-scale computation are as follows (refer to Figure 1):

- **Step 1** Trace image intensity along a set of pairs of radially opposite sample lines emanating from *p*.
- Step 2 Locate the closest edge point on each sample line (triangles and black dots).
- **Step 3** Reposition the edge locations on each pair of opposite sample lines according to the axial symmetry of an ellipse (black dots to white dots).
- **Step 4** Compute the t-scale at *p* using the best-fit ellipse derived from the repositioned edge points (triangles and white dots).

It may be mentioned that basic steps for t-scale computation adopted here are same as presented in [20] and Figure 1 is a modified version of Figure 1 in [20]. In this paper, we report significant improvements of the algorithms employed in individual steps of t-scale computation. Edge detection (Step 2) and ellipse fitting (Step 4) are the two crucial steps in t-scale computation framework that largely determine the overall performance and accuracy of the method. Often, image noise and blurring cause errors in edge detection leading to artifacts in local structural definition represented by t-scale. Here, we present a new optimization scheme to determine edge detection kernels for given kernel size and edge parameters. For the ellipse fitting step, algebraic and geometric approaches are combined in a synergistic manner leading to an improved solution within the constraints of computational complexity. Steps 1 and 3 are straightforward and the original solutions by Saha *et al.* [20] are adopted here.

At this point it may be clarified that, although the medical image interpolation addressed here is a 3-D problem, it is solved by using structural correspondence between every two successive image slices. Here, the role of t-scale is to provide this structural correspondence which is solved in 2-D. Therefore, we describe t-scale computation in 2-D only. However, the method immediately generalizes to 3-D with appropriate changes in Steps 1 and 4. In 2-D, profiles are intended to be uniformly distributed over 2-D angular space around a candidate image point.

Step 1: Intensity computation along sample lines—We select *m* pairs of mutually opposite sample lines at an approximately uniform angular distribution ensuring that the computed t-scale is not skewed in one direction. Three parameters are identified with sample

lines, namely, the number of sample lines, the length of each sample line, and the interval length between two successive sample points on each line. Following our experience, we recommend using ten to fifteen pairs of sample lines. The length L of a sample line represents the largest distinguishable local scale/structure size; i.e., the extent of locality or neighborhood. Obviously, too small a value of sample line length is not desirable; again a very large value of this parameter adds to the computational burden while the additional information may not be so important. Therefore, a correct choice of the sample line length parameter is important and application dependent. The sample interval length δ between two successive points on a sample line is determined by the trade-off between the computational complexity and the scale of the finest detectable structure. Assuming that the image resolution is appropriate for the target application, we recommend choosing δ between '1' and '0.5' times the smallest dimension of a pixel. Following the Nyquist theorem of sampling, $\delta = 0.5$ is the sufficient condition for reconstruction of a signal and therefore, any $\delta < 0.5$ will bring no additional benefit. On the other hand, if $\delta > 1$ is used, we clearly miss one voxel thick structures in an image. The intensity at any sample points is determined using bilinear interpolation of the four binding image points. As we mentioned earlier, tscale is computed over individual image slice and therefore, a sample point always falls inside a box bounded by four image points. Let $f_i^p(v)$ denote the intensity computed at the v^{th} sample point on the i^{th} sample line emanating from a pixel p.

Step 2: Improved edge location on each sample line—To eliminate the effects of locally disconnected structures of similar intensities during edge detection, two connected-intensity profiles μ_{UP} and μ_{DN} are derived from the intensity values $f_i^p(v)|v=0, 1, 2, ...$ along the *i*th sample line:

$$\mu_{UP}(v) = \max_{x=0,1,\cdots,v} f_i^P(x),$$
(1)

$$\mu_{DN}(v) = \min_{x=0,1,\cdots,v} f_i^p(x).$$
(2)

The above two connected-intensity profiles along a sample line significantly eliminate the effects of locally disconnected structures intersecting with the sample line without causing any blurring and thus preserving thin structures (Figure 2). In Figure 2, the connected-intensity profile μ_{UP} successfully eliminates effects of locally disconnected structures of bright intensities. However, it fails to identify the step down edge which may be detected with μ_{DN} only. The purpose of using the two different connected-intensity profiles is to separately handle the cases of "step-up" and "step-down" edges.

Conventional Laplacian of Gaussian or LoG [32] is used for detecting edges on the intensity profile along a sample line. An edge is located at the first zero crossing of the LoG with its strength of intensity gradient, computed using a derivative of Gaussian or DoG kernel, exceeding a predefined gradient threshold. A critical issue with this approach is how to determine optimum discrete kernels for the LoG and DoG operations under given constraints. Gunn [33] has reported a task- and image-dependent approach of optimizing a discrete kernel for LoG. However, a kernel optimized for one image may not necessarily be optimal for another image. Here, we develop a task- and image-independent optimization approach for computing discrete LoG or DoG kernels that is based on minimizing total discretization errors under given constraints of kernel length and edge characteristics. Essentially, a discrete kernel is computed from the corresponding function by sampling it at a regular interval Δ generating 2N + samples for a given value of N as illustrated in Figure

3(a). We will use LoG(x), where $x \in \mathbb{R}$, to denote a continuous LoG function and LoG[n], where $n \in \{-N, \dots, 0, \dots, N\}$, to denote the sample value of the continuous LoG function at $n\Delta$, i.e., $LoG[n] = LoG(n\Delta)$. To avoid any confusion, we will use "LoG" as an abbreviation of "Laplacian of Gaussian" while the italicized LoG to denote a LoG function. An important observation here is that, although, $\int LoG(n)dx = 0$, due to the finite length of the kernel, the sum of positive sample values over $LoG[n]|_{n=-N,\dots,-1,0,1,\dots,N}$ may not equate the sum of

sum of positive sample values over $LoG[n]|_{n=-N,\dots,-1,0,1,\dots,N}$ may not equate the sum of $\sum_{n=-N}^{N} LoG[n]$ negative values leading to a nonzero value for n=-N. This discrepancy may lead to artifactual shifts in computed zero crossings. This challenge is overcome by scaling the sample values as follows:

$$dLoG[n] = \begin{cases} \frac{1}{s} \times LoG[n], & LoG[n] > 0, \\ s \times LoG[n], & LoG[n] < 0, \end{cases}$$
(3)

where

$$s = \sqrt{\frac{\sum_{n=-N}^{N} |LoG[n]| + \sum_{n=-N}^{N} LoG[n]}{\sum_{n=-N}^{N} |LoG[n]| - \sum_{n=-N}^{N} LoG[n]}}.$$

Note that s^2 is the ratio of the sum of positive LoG sample values to that of negative values. It is worth mentioning that such scaling factor is not needed for the computation of DoG kernels.

As illustrated in Figure 3(b), a discretization process with a fixed kernel length is always associated with two types of errors and we refer to these errors as *step error* and *truncation error*. Note that the scaled sample values are used in the final kernel and therefore the errors are computed after scaling. It may also be noted from Figure 3(b) that the truncation error monotonically decreases with sample interval size Δ ; however such monotonicity property is absent for the step error primarily due to the scaling factor. The optimization algorithm determines the sample interval size Δ that minimizes total discretization error which is the sum of step and truncation errors and uses the sample interval size to compute the kernel. For all experiments presented in this paper, the LoG and DoG kernels are optimized separately. Further, it may be mentioned that if a sample line hits an image boundary without finding an edge point, the algorithm enforces an edge at the end of the sample line.

To evaluate the performance of our kernel optimization algorithm, we have performed the following experiment. We computed an ideal LoG kernel LoG_{ideal} by densely sampling the LoG function. Specifically, we represented LoG_{ideal} by 10001 uniform samples covering $\pm 5 \times \text{sigma}$ of the analytical LoG function leaving only $\sim 10^{-7}$ % unaccounted energy of the function. For a given kernel size 2N + 1 and a given sample interval size Δ , a discrete LoG kernel $LoG_{N,\Delta}$ was constructed by sampling a LoG function on both sides of zero. One hundred sample lines were randomly selected from each of the sixteen images used in our experiments resulting in 1600 sample lines. At each point *t* on a sample line *l*, both ideal and discrete LoG maps, denoted by l_{ideal} and $l_{N,\Delta}$, respectively, were computed by convolving with LoG_{ideal} and $LoG_{N,\Delta}$. Finally, the error function $\text{Error}_{N,\Delta}$ was computed from all sample lines as follows:

$$\operatorname{Error}_{N,\Delta} = \sum_{l} \sum_{n} \frac{|l_{\operatorname{ideal}}(n) - l_{N,\Delta}(n)|}{l_{\operatorname{ideal}}(n)}$$

In the above equation of error computation, we used the central two-third points on each sample line to avoid edge effects in LoG computations. Results of experimental error for kernel size of nine at different sample interval Δ are presented in Figure 3(b). It may be observed from the figure that the theoretical error function is notably different from the experimental error function. Despite the difference in two error functions, the theoretical optimum value for Δ is close to the experimental optimum value for the same. Also, we performed the same experiment for different kernel sizes and the results of optimum theoretical and experimental sample interval sizes are presented in Table 1. It may be observed that for an extremely small kernel size (3 or 5), the difference between the theoretical and experimental optimum sample sizes is relatively high and it may be caused by instability of a discrete LoG kernel due to insufficient number of samples at extremely small kernel size optimized for one image may not be optimum for another image supporting our image-independent optimization approach.

Step 3: Repositioning of edge points—The edge points obtained in the previous step are intended to roughly describe the boundary of the t-scale ellipse centered at the candidate image point p. Following the axial symmetry of an ellipse, for each pair of opposite sample lines, the two edge points should be equidistant from p, which is the center of the t-scale ellipse. However, the detected edge points on a local structure do not necessarily satisfy this property. For example, in Figure 1, the edge locations marked as black dots on the northbound sample lines are mostly farther from p than the edges on corresponding opposite sample lines. Therefore, the edge points need to be repositioned by analyzing the edge points on every pair of sample lines. Specifically, between the two edge points on a pair of sample lines, the one closer to the candidate image point p is selected and reflected on its complementary sample line. The edge points marked as white dots in Figure 1 are obtained using this repositioning scheme.

Step 4: T-scale ellipse fitting—The final step in t-scale computation is to fit an ellipse to the repositioned edge points obtained by Step 3. In our application, the number of edge points is significantly larger as compared to the number of parameters needed to represent an ellipse. The literature on ellipse fitting is quite mature and there are several established approaches available in the literatures [34–36] to accomplish the task. All ellipse fitting approaches essentially minimize the error between the observed data (here, the edge points) and the computed ellipse. Primarily, these methods differ with respect to the definition of these errors. We have investigated two different distance measures defining these errors, namely, algebraic and geometric distances [37]. As mentioned by Gander et al. [37], the algebraic approach to ellipse fitting suffers from stability and often fails to provide a "good looking" result. These problems of the algebraic approach are enhanced with the increase of anisotropy of an ellipse and such situations may occur frequently in t-scale computation, especially, when a candidate image point is close to an edge. However, the algebraic approach for ellipse fitting leads to a canonical solution and is well-known for its efficiency. On the other hand, a geometric distance approach generates stable solutions for most ellipses and also uses more natural Euclidean distance metric. Unfortunately, it is difficult to derive a closed form solution for geometric distance minimization in ellipse fitting, and therefore, a geometric distance based approach is commonly realized using a stochastic search process

raising the issue of initialization. The primary motivation behind our ellipse fitting algorithm is to utilize the benefits of both algebraic and geometric approaches. Here, we use the algebraic distance based solution for initialization to the geometric distance based approach providing that the former method yields a real ellipse. Under the situation where algebraic distance fails, principal component analysis (PCA) of repositioned edge points is used to obtain the initial solution. In the following algorithm for optimal ellipse computation, edge points refer to repositioned edge points.

- **Step 1** Translate all edge points so that the candidate image point *p* is moved to the origin.
- **Step 2** Compute the covariance matrix of the translated edge points and compute its eigenvectors \mathbf{i}_1 and \mathbf{i}_2 and the eigenvalues λ_1 and λ_2 .
- **Step 3** Rotate the edge points around the origin so that \mathbf{i}_1 and \mathbf{i}_2 are aligned with coordinate axes. Solve canonical equations [20] to compute the ellipse that minimizes algebraic distance errors.
- **Step 4** If the ellipse computation in Step 3 is a real one, use it for Step 5; otherwise, use the ellipse with semi axes $\lambda_1 \mathbf{i}_1$ and $\lambda_2 \mathbf{i}_2$.
- **Step 5** Compute the final ellipse by minimizing geometric distance errors from edge points with the initialization obtained in the previous step. Newton's algorithm along with the Jacobian of the error function is iteratively used to obtain the optimization of the target ellipse.

Figure 4 illustrates the final ellipse fitting with both PCA and algebraic distance based initialization. It can be noticed that both initializations lead to the same ellipse. However, the initialization with the algebraic distance based solution leads to a faster convergence. To evaluate the performance of our ellipse-fitting algorithm, we performed an experiment on ten thousand ellipse datasets randomly selected from our t-scale computation experiments on medical images. Specifically, we applied the stochastic optimization algorithm for geometric distance between the central candidate point and its closest repositioned edge point, (2) same as (1) but for the farthest point, (3) same as (1) but for the average distance, (4) ellipse computed by PCA, and (5) ellipse computed using algebraic approach. In our experiment, the algebraic approach failed to provide a valid solution for approximately 1% datasets and we used PCA-based initialization for such data. Average numbers of iterations required for five different initializations were: 24.5, 21.3, 18.5, 14.9, and 11.8, respectively.

Let us use $\iota_1(p)$ and $\iota_2(p)$ to denote the major and the minor semi axes of the t-scale ellipse at an image point p. It may be noted that the length of the minor semi axes $|\iota_2(p)|$ alone represents local scale at the image point p using a spherical model [13]; we will refer to it as the *isotropic scale* of p.

T-scale Visualization—To display a t-scale image, we apply a HSI color coding scheme following the fact that an ellipse centered at origin is uniquely defined by three factors. Specifically, the color at an image point p is defined from its t-scale as follows: hue =

orientation of the major semi axis $\mathbf{\iota}_1(p)$; saturation anisotropy= $\sqrt{1 - |\mathbf{\iota}_2(p)|^2/|\mathbf{\iota}_1(p)|^2}$, and intensity = normalized thickness = $|\mathbf{\iota}_2(p)|/S_{length}$, where S_{length} denotes the length of sample lines used for t-scale computation. Results of application of both the previous method [20] (here onward, we will refer to this method as the "old" method) and the new method for tscale computation to a sagittal slice from an MR brain data are presented in Figure 5(c) and (d) respectively. In both images, the HSI color coding scheme of Figure 5(b) are used to represent t-scales at individual points. It may be noted from Figure 5(c) and (d) that although

the two color-coded t-scale images are visually similar, the result using the new method is significantly cleaner with less noise (spurious dots).

2.2 T-scale Based Image Interpolation

The general purpose of medical image interpolation is to increase image resolution along the slice direction using an image post-processing algorithm. Although classical interpolation methods use neighborhood intensity analysis techniques over adjacent slices, more advanced algorithms follow the approach of detecting "interpolation lines" [38] using the correspondence of local structures in adjacent slices. Here, we present a new interpolation method that provides a closed form solution for identifying this correspondence using its tscale information in adjacent slices, which is used to compute local deformation as well as the interpolation line at the individual image point. The key idea is to compute the local deformation between two adjacent slices from their t-scale information which is accomplished in two steps -(1) computation of t-scale-derived normal vector (Figure 6(a)) associating each image point on a slice to its nearest edge point and (2) determine local deformation between two adjacent slices using a closed form equation involving normal vectors in two adjacent slices (Figure 6(b)). Let p_A and p_B be the spatially matching image points on two adjacent slices, namely, sl_A and sl_B ; thus p_A and p_B have the same twodimensional (2D) coordinates, say p, in the two slices. Let N_A and N_B be the normal vectors at p_A and p_B , respectively. It is interesting to observe from the figure that the deformation $\tau(p, sl_B, sl_A)$ from sl_B to sl_A at the 2D image point p may be determined by subtracting the vector N_B from N_A , i.e.,

$$\tau(p, sl_B, sl_A) = \mathbf{N}_A - \mathbf{N}_B. \tag{4}$$

In the following, we first describe the method of computing normal vectors from local tscales and then explain the interpolation method using normal vectors. Finally, we discuss the challenges with the method and offer their solutions.

2.2.1 Computation of normal vectors using t-scale—As mentioned earlier, the purpose of normal vector is to represent the nearest edge point associated with each image point and its direction is orthogonal to the orientation of the local structure (Figure 6(a)). Following the fact that t-scale represents the orientation of the local structure along with its size in different directions, the association between normal vector and t-scale is obvious. Specifically, the normal vector is related to the minor semi-axis of the t-scale at an image point *p*. However, as illustrated in Figure 7(a), an ambiguity arises as the t-scale of a point *p*.

fails to directly indicate which of the two vectors \overrightarrow{pb} and $\overrightarrow{pb'}$ points to the nearest boundary. This ambiguity is solved by analyzing t-scale-derived isotropic scale along the direction of

minor axis bb'. As illustrated in Figure 7(b), isotropic scales reduce along the vector \overrightarrow{pb}

pointing to the local boundary; therefore, \overrightarrow{pb} represents the normal vector at *p*. The color coded representation of normal vector map for the image of Figure 8(a) is presented in Figure 8(e).

2.2.2 Interpolation algorithm—In Section 2.2, we have described the basic principle of computing local deformation using t-scale derived normal vectors. Here, we describe the new interpolation algorithm. Let us consider two image slices sl_A and sl_B separated by a distance of Δ_{sl} as shown in Figure 9(a) and we want to interpolate the intensity at a <u>target</u> point $p = (x, y, t_0)$ in between the two slices. Here, the location t_0 of p in the z-direction is defined with respect to the location of the base slice sl_B ; thus, $0 < t_0 < \Delta_{sl}$. Also, we assume

that *x* and *y* are integer valued. The interpolation method essentially computes the interpolation line $l(t)_{0 \le t \le \Delta_{sl}}$ passing through the target point *p*; the line l(t) gives three dimensional coordinates of a point at the height of *t* from the base slice. Once the interpolation line l(t) is determined, the two points c_A and c_B at the intersection of the line with respective slices are computed and the intensity at *p* is determined using linear interpolation of the intensities at c_A and c_B . Thus, to complete the algorithm, we need to describe the method of computing the interpolation line l(t). First, the projections p_A and p_B of the point *p* on the two slices sl_A and sl_B are determined; thus, $p_A = (x, y, \Delta_{sl})$ and $p_B = (x, y, 0)$. Let \mathbf{N}_A and \mathbf{N}_B be the normal vectors at p_A and p_B with b_A and b_B being the nearest points on respective local boundaries as illustrated in Figure 9(a). It may be noted that the line l'(t) obtained by joining the two points b_A and b_B results in a local 2D deformation of $\mathbf{N}_A - \mathbf{N}_B$ along the slice plane when the image slice sl_A warped onto sl_B ; let $\mathbf{N}_{AB} = \mathbf{N}_A - \mathbf{N}_B$. However, the line l'(t) may not pass through the point *p*. Therefore, the interpolation line is computed by appropriately shifting the line l'(t) along the slice plane so that it passes through *p*. The final interpolation line is computed using the following equation:

$$l(t) = \begin{bmatrix} x \\ y \\ t_0 \end{bmatrix} + (t - t_0) \times \begin{bmatrix} \mathbf{N}_{AB} \cdot \mathbf{i}_{\mathbf{X}} \\ \mathbf{N}_{AB} \cdot \mathbf{i}_{\mathbf{Y}} \\ \Delta sl \end{bmatrix}.$$
(5)

In the above equation, \mathbf{i}_x and \mathbf{i}_y denote the two unit vectors along the x- and y-coordinate axes, respectively. It may be interesting to note that, if the target point p is moved close to lower slice sl_B at (x, y, t_{low}) as illustrated in Figure 9(b), the initial line l'(t) is not changed. However, in this situation, we need to translate the line l'(t) differently so that it passes through the target point, which is now closer to slice sl_B . Thus, the final interpolation line l(t) is changed as illustrated in Figure 9(b); note that in Eq. 5, t_0 should be replaced by t_{low} . Since, l(t) intersects the line $p_A p_B$ at p, the point c_B should be close to p_B . Again, on the interpolation line l(t), the target point p is much closer to c_B than to c_A . Thus, following linear interpolation, the interpolated intensity at p should be very close to that of the point c_B which is again close to the point p_B . The method works similarly when the target point p is moved close to the upper slice sl_A . Further, it may be pointed out that the shifting of initial interpolation line l'(t) to l(t) to determine the final interpolation line at p essentially assumes that deformation vectors over a small neighborhood are similar and ignores local variations in the deformation field. This assumption is made by most registration-based medical image interpolation algorithms [30] where the deformation field is represented using a smooth function. It may be noted that, in our algorithm, the shift of an interpolation line is always smaller than one voxel in the slice direction.

2.2.3 Algorithm: Challenges and Their Solutions—As discussed above, the basic principle of the method lies in identifying the nearest boundary point at each image point which is represented by its normal vector. A major challenge with this approach is that the normal vector is less stable near the medial axis of a local structure as it may point to any of the two opposite boundaries of the local structure leading to an ambiguity. Depending upon which of the two opposite boundary points is selected, the normal vector will significantly differ thus being a source of errors in the computation of deformation vectors used for interpolation.

This problem is overcome by detecting local medial points and taking a special care at those points while computing the deformation vector or the interpolation line. The medialness at an image location is determined by comparing its isotropic scale with another t-scale-derived feature indicating the "local structure width". At any image point p, the *local structure width* denoted by *LSW*(p) represents the width of the local structure around p and

is defined as twice the closest maximal isotropic scale along the direction of the t-scale minor axis **iota**;₂(p). Figure 10 illustrates the local structure width map for a phantom image computed using the above definition. Finally, the *medialness* of an image point p, denoted by M(p), is defined as follows:

$$M(p) = \begin{cases} 1, & \text{if } 2|\boldsymbol{\iota}_2(p)| \ge LSW(p), \\ \frac{2|\boldsymbol{\iota}_2(p)|}{LSW(p)}, & \text{otherwise.} \end{cases}$$
(6)

Here, image points with a medialness value greater than or equal to 0.75 are considered as medial points which are treated separately to avoid the errors due to the ambiguity outlined above. Selection of a threshold close to '1' makes the algorithm vulnerable to noise. On the other hand, a lower value of the threshold leads to exclusion of image points. Here, we have used a threshold of 0.75 to ensure that 75 percent of image points may directly be used in the interpolation algorithm. To solve the problem of ambiguity at medial points, we assume that that the displacement of a structure between two adjacent slices is less than half of the local structure width. Let us consider the situation of Figure 11 where the projection of the interpolation point p on the slice sl_A is a medial point and let a and a' be the nearest points on the two opposite edges of the structure around p_A . It may be noted from the figure that the point a on the slice sl_A corresponds to the edge point b on the matching structure around p_B on the slice sl_A . Due to the ambiguity near the medial point p_A , it is difficult to say whether $\overrightarrow{p_A a}$ or $\overrightarrow{p_A a'}$ is the true normal vector. However, with the assumption that the local deformation is less than half of the structure width, the magnitude of the deformation using

whether $p_A a$ or $p_A a$ is the true normal vector. However, with the assumption that the local deformation is less than half of the structure width, the magnitude of the deformation using the correct correspondence of a and b must be less than that of the wrong correspondence of a' and b. Therefore, the 2D deformation N_{AB} is computed using the following equation

$$\mathbf{N}_{AB} = \begin{cases} \mathbf{N}_{A} - \mathbf{N}_{B}, & \text{if} |\mathbf{N}_{A} - \mathbf{N}_{B}| \leq |-\mathbf{N}_{A} - \mathbf{N}_{B}|, \\ -\mathbf{N}_{A} - \mathbf{N}_{B}, & \text{otherwise.} \end{cases}$$
(7)

The situation where the projection point p_B is a medial point may be solved similarly. Finally, a Gaussian smoothing filter of kernel size 5×5 is applied on the computed deformation field to capture larger contextual information.

3 EXPERIMENTAL METHODS

In this section, we describe our experimental approach to examine the performance of the improved t-scale computation algorithm and the new t-scale based image interpolation method. Specifically, we quantitatively examine the robustness of the improved t-scale computation method at various levels of noise and blurring and compare its performance with that of the old method [20]. For image interpolation, we evaluate the accuracy of the t-scale based method and compare the performance with current state-of-the-art methods.

3.1 Robustness of t-scale computation at varying noise and blurring

Performance of the improved t-scale computation has been examined both qualitatively and quantitatively. The experimental setup of [20] has been followed in this paper to quantitatively evaluate the robustness of t-scale computation under varying levels of noise and blurring. For this purpose, a set of 250 realistic 2D phantom images at five different levels of noise and five different levels of blurring were generated from manual segmentations of white matter regions in image slices from ten different MR head data sets

using 3DVIEWNIX [39] graphical interface; see Figure 12 for examples. Let $C^{T_i}|_{i=1,2,\dots,10}$ denote a true phantom image and let $C_{_{RN}}^{T_i}$ denote the phantom image obtained from e^{τ_i} at the level of blurring *B* and noise *N*. Also, let $\Gamma^{T_i}(p)$ (or, $\Gamma^{T_i}_{_{RN}}(p)$) denote the t-scale ellipse computed at an image point p in the image \mathcal{C}^{T_i} (respectively, $\mathcal{C}_{BN}^{T_i}$). In order to evaluate the robustness of a t-scale computation method under noise and blurring, $\Gamma^{T_i}(p)$ is considered as the truth because it is computed from \mathcal{C}^{τ_l} with no noise and blurring. A "similarity measure" of two concentric ellipses is crucial to compute the robustness of a t-scale computation algorithm. Unlike scalar and vector quantities, similarity between two ellipses (tensor) is not trivially defined. Here, we have utilized the natural mapping (see Section 2.1) that exists between the set of all ellipses and the RGB color space. Specifically, the disagreement between two ellipses is defined as the difference between their representative RGB vectors (the range of each component of RGB vector is [0, 1]). Let **RGB**($\Gamma^{T_i}(p)$) and **RGB**($\Gamma^{T_i}_{_{RN}}(p)$) denote the RGB vectors onto which the ellipses Γ^{T_i} and, $\Gamma^{T_i}_{_{RN}}(p)$ respectively, are mapped. Thus, $(1/\sqrt{3}) | \mathbf{RGB}(\mathbf{\Gamma}^{T_i}(p)) - \mathbf{RGB}(\mathbf{\Gamma}^{T_i}_{BN}(p)) |$ captures the difference between true and computed t-scales at p in terms of their distance in the color-space used here to represent tscales. Here, a division by $\sqrt{3}$ is applied to normalize the error measure so that its value lies in the [0,1] interval. The figure of merit $FOM_{BN}^{T_i}$ (a measure of similarity) of a t-scale computation method for a phantom image $C_{BN}^{T_i}$ is defined as follows:

$$FOM_{BN}^{T_{i}} = \frac{\left\| C^{T_{i}} \right\| - \sum_{p \in C^{T_{i}}} \left(1/\sqrt{3} \right) |\mathbf{RGB}(\boldsymbol{\Gamma}^{T_{i}}(p)) - \mathbf{RGB}\left(\boldsymbol{\Gamma}_{BN}^{T_{i}}(p)\right)|}{\left\| C^{T_{i}} \right\|} \times 100.$$
(8)

In the above equation, $|| \mathcal{C}^{T_i} ||$ denotes the number of pixels in \mathcal{C}^{T_i} and $|\cdot|$ returns the absolute value of its parameter. For a given method, at a given level of noise and blurring, the mean and standard deviation of $FOM_{BN}^{T_i}|_{i=1,2,\cdots,10}$ values have been computed. Also, at a given level of noise and blurring, a paired t-test has been performed on the $FOM_{BN}^{T_i}|_{i=1,2,\cdots,10}$ values obtained by the old [20] and the new methods.

3.2 Accuracy of t-scale based image interpolation

The accuracy of the t-scale based interpolation method has been examined both qualitatively and quantitatively on phantom and real images and has been compared with a current state-of-the-art registration based method [38]. For the registration based method, both B-spline and Demons registration techniques, implemented in *ITK* [40], were used for comparison. For the B-spline registration based algorithm, we used the parameter setting recommended in [38]. Also, for both the B-spline registration based and Demons registration based algorithms, we used the stopping criterion as minimization of the residual errors for known truths. Thus, we experimental results presented here represent the optimum performance for the two methods in terms of iteration number. On the other hand, for the new method, no such stopping criterion or iteration number is needed.

Quantitative evaluation experiments were designed following the conventional approach [38] of evaluating an image interpolation approach. Specifically, for a given 3D image consisting of k uniformly spaced slices, each slice except the first and the last one was removed one at a time and was regenerated from its two neighboring slices by a given method of interpolation. The error of the specific interpolation method was computed by comparing the original slice and the one obtained by interpolation. The mean absolute difference was used to measure the accuracy of the corresponding interpolated slice. Let

 $I_{original}(x, y, i)$ and $I_{interpolated}(x, y, i)$, respectively, denote the original and the interpolated image intensities at the in-plane image location (x, y) on the *i*th slice of image of size $m \times n \times k$. The interpolation error for the *i*th slice, denoted as MAD_i , is computed as follows:

$$MAD_{i} = \frac{1}{mn} \sum_{x=1}^{m} \sum_{y=1}^{n} |I_{original}(x, y, i) - I_{interpolated}(x, y, i)|.$$
(9)

Finally, the interpolation error over the entire image, denoted by MAD, is computed as:

$$MAD = \frac{1}{k - 2} \sum_{i=2}^{k-1} MAD_i.$$
 (10)

In our experimental study, we have used both phantom and real medical images. In the following, we describe each of these data sets.

3.2.1 Simulated brain phantom data—The phantom data set was generated from the simulated brain MRI data from the BrainWeb site (http://www.bic.mni.mcgill.ca/brainweb). Specifically, the T1 weighted MR image (in plane size: 181×217 ; number of slices: 181) of anatomical model for normal brain at 0% noise, 0% intensity non-uniformity and $1 \times 1 \times 1$ mm³ voxel size was used. Two sets of test phantoms were generated from this simulated data. The first dataset was created by adding correlated white Gaussian noise at different SNR values while the other dataset was produced by adding different levels of intensity inhomogeneity. Ten different SNR values ranging between 50 and 10 were selected for the experiment. To examine the robustness of the method with respect to the intensity inhomogeneity, a multiplicative Gaussian inhomogeneity model was used with its center, height, and width chosen randomly. Altogether, fifteen images at different intensity inhomogeneity levels with the width of the inhomogeneity Gaussian varying between 68 and 132 mm and the height varying between 10% and 50% of the image intensity range were used for the experiment. Specifically, three Gaussian inhomogeneity images were randomly generated at varying heights, widths, and locations of the center. Each of these initial inhomogeneity maps was multiplied by a constant α and added to the original image to obtain a test image at a given percentage of intensity inhomogeneity as defined by the following equation:

percentage of inhomogeneity =
$$\alpha \times \frac{\text{max inhomogeneity intensity}}{\text{max image intensity}}$$
. (11)

For a given percentage of inhomogeneity and a given initial inhomogeneity map, a test image was generated and was used for the experiment. For a given method, the error MAD_i was computed for each slice i = 2, 3, ..., k - 1 in the test image using Equation 9. Since we used three initial inhomogeneity maps, we got three measures of MAD_i for the *i*-th slice and a given method at a fixed percentage of inhomogeneity; an average of these three errors was used to compute MAD (Equation 10) and for paired t-test while comparing with results from another method.

3.2.2 Medical Images—Sixteen 3D images from different body regions and different clinical applications were used to examine the performance of our method on real data. Our first data set consists of five abdominal CT datasets from five different subjects with voxel size: $0.59 \times 0.59 \times 1.00 \text{ mm}^3$ and in-plane grid size: 512×512 with the number of slices

varying between 64 and 319. For convenience, we will refer to these images as Abdomen1, Abdomen2,...,Abdomen5, respectively. Our next data set included CT images of four cadaveric ankle specimens scanned with a Siemens Sensation 64 Multi-slice CT scanner at 120 kVp and 140 mAs (voxel size: $0.21 \times 0.21 \times 0.3$ mm³) adequately displaying trabecular bone micro-architecture. For these images, the common in-plane image grid size was $512 \times$ 512 and the number of slices varied between 334 and 336. We will refer to these images as Ankle1, Ankle2, Ankle3, and Ankle4, respectively. The last group consisted of seven pulmonary CT images from seven different subjects scanned on a Siemens Sensation 64 Multi-slice CT scanner at 120 kVp and 200 mAs yielding $0.55 \times 0.55 \times 0.5$ mm³. The inplane grid size for these images was 512×512 and the numbers of slices varied between 519 and 728. These images will be referred to as Lung1, Lung2, ..., Lung7. Ankle and lung data sets were chosen to evaluate the performance of an algorithm under changing topology and shapes of micro-structures along the slice direction. For each of the phantom as well as medical data, the overall interpolation error was compared by analyzing their MAD values. The errors on individual slices were computed for different methods and paired t-tests were performed to evaluate statistical reliability of differences in overall interpolation errors by different methods. The *p*-value of 0.05 was considered significant.

4 RESULTS AND DISCUSSION

In this section, first, we discuss the results of the experiments described in Section 3.1 comparing the performance of the new t-scale computation algorithm reported here with the old method [20]. Results of application of the old and new t-scale computation algorithms on a sagittal MR image of human head are presented in Figure 5 and qualitatively compared in a zoomed-in region. Improvements obtained by the new method are visually apparent. Results of application on a phantom image referred to in Section 3.1 at low, medium and high levels of noise and blurring are illustrated in Figure 12. Here too, the new method appears to show more robustness and to better preserve structural information at different levels of noise and blurring. Table 2 presents the mean (first entry) and standard deviation (second entry) of FOM (figure of merit) values for different phantom images referred to in Section 3.1. Here, each cell presents the statistics of ten phantom images at a specific level of noise and blurring with noise increasing from top to bottom while blurring increases from left to right. Table 3 presents the results of comparison between the new method and the old method reported in [20] for ten images at different levels of noise and blurring. It may be noted from Table 3 that the new method outperforms the old method at all levels of noise and blurring except for three cases (N1, B5), (N2, B5) and (N3, B4). The p-values of paired t-tests for these cases indicate that the differences between the old and new methods are not statistically significant. The performance of the new method is statistically significantly better in all other tested cases (p < 0.01). It may also be noted from the table that the margins in FOM values between the two methods increases with the increase of noise suggesting increased relevance of the new method at higher noise levels. The exceptional behavior of the three cases in our comparative experimental results may be explained by the fact that, at low noise and extremely high blurring, the difference in performance of the two methods is reduced. High blurring leads to high uncertainty in edge location and both methods suffer from this uncertainty. On the other hand, low noise levels give only a little room to the new method for improvement resulting in statistically equal performance. In the current paper, we have not focused on the computational efficiency of t-scale computation. The original algorithm for t-scale computation takes 32 seconds for a 256×256 image using a 2.53 GHz Intel(R) Xeon(R) CPU running under Linux OS. The improved t-scale computation method presented here takes 53 seconds for the same image running on the same machine. The extra time is mostly consumed by the improved ellipse fitting algorithm.

Here, we present results of the experiments described in Section 3.2.1 to compare the new interpolation method with two registration based methods using B-spline and Demons techniques at various levels of added noise and intensity inhomogeneity on the BrainWeb phantom dataset. Results of quantitative analysis for three different methods at various noise levels are presented in Figure 13. At every level of noise, the t-scale based interpolation method has outperformed the demons registration based method and the results are found to be statistically significant using a paired t-test. As compared with the B-spline registration based method, the new method has outperformed (p < 0.01) at all levels of noise except at SNR = 10, 12, 14, and 16 where interpolation errors by the two methods are not statistically different. These results indicate, that although the t-scale based method outperforms the two registration based methods, the overall difference in performance is reduced with increasing noise. A possible argument behind the observation is that the closed form equation for the interpolation line in t-scale based approach uses the t-scale of only two points, one from each adjacent slice. Thus, the approach has less strength for statistical noise smoothing as compared to the two registration-based methods and the performance of the current t-scale based registration method on high noisy images may be a limitation. A possible avenue for improving the performance of the t-scale based method may be to amalgamate the closed form solution into the registration framework and a research effort is currently ongoing in our laboratory along this direction.

Comparative results of the three methods at various levels of image inhomogeneity are presented in Figure 14. It may be noted from these results that effects of image inhomogeneity for all three method are minimal which was expected as intensity inhomogeneity has only limited effects on local structural information in an image. Also, it is important to note that at all examples of inhomogeneity, the t-scale based interpolation method has outperformed the two registration based methods (p < 0.001). Here, it may be observed from Figure 13 and Figure 14 that the standard deviation values are somewhat high as compared to the corresponding differences of MAD values. Yet, the p-values indicated statistical significance of the measured differences in errors which is associated with a proper use of a paired t-test. The results of these experiments suggest that although there are significant variations in interpolation errors from one slice to another, the differences in performance for two methods are consistent from one experiment to another. Results of applications of the three methods on several medical images are illustrated and visually compared in Figure 15-Figure 17. In each of these figures, the top row show three consecutive image slices from the specific dataset. The middle row presents the interpolated results for the central slice of the top row computed by three different methods using the left and right image slices of the top row. The last row indicates the absolute error maps for corresponding interpolated result as compared with the original image slice. In the middle and last rows, the images on the left and central columns represent the results of B-spline and Demons based registration based methods. The right column presents the results of the t-scale based method. In all these examples, reduction of interpolation errors by the t-scale based method is visually notable. For the example of Figure 15, it may be noted that all methods including the t-scale based algorithm produce high concentration of interpolations errors in the trabecular bone region. These errors are primarily contributed by the highfrequency changes in trabecular bone topology from one slice to another. Similar to other algorithms, t-scale based registration method may fail to compensate for large topological variations from one slice to another. A quantitative analysis of the results is presented in Figure 18 which indicates the mean and standard deviation of interpolation errors by three methods for different medical images described in Section 3.2.2. The mean error was computed as the MAD value over the entire image while the standard deviation of errors was computed as the standard deviation of MAD_i values from individual slices (see Section 3.2). As observed in this figure, the t-scale based method has always outperformed the other two methods with a clear margin. It may be interesting to observe from the quantitative results

that, across all data and methods, the standard deviation values are small as compared to corresponding mean error. It indicates that the interpolation error is a highly reliable parameter to determine the quality of individual methods. Also, a paired t-test of errors over individual image slices was performed between t-scale based methods and each of the two registration based methods. The results of every paired t-test comparing the performance of the t-scale and another method demonstrated the superiority of the t-scale method at high significance levels (p-value < 0.001), see Figure 18. Further, it may be observed that errors for ankle and pulmonary images are relatively larger as compared to abdominal images. Possibly, it is caused by rapid changes in both topology and geometry of trabecular bone in ankle images and both airways and vasculature in pulmonary images. As for the computation time, the speed of B-spline and Demons registration largely depends on the image size and the minimization procedure; on the other hand computation time for t-scale computation mostly depends on image size and the parameter setting for t-scale computation, i.e., the number and length of sample lines. For all above experiments, we have used 90 sample lines with each sample line being 50 pixel long. In the following, we present the average computation time of the three methods for different real image sets used in our experiments:

Lung images: B-spline (12 minutes), Demons (8 minutes), and t-scale (10 minutes)

Ankle images: B-spline (9 minutes), Demons (7 minutes), t-scale (10 minutes)

Abdomen images:, B-spline (10 minutes), Demons (3 minutes), t-scale (9 minutes).

5 CONCLUSION

An improved t-scale computation method has been presented and its application to medical image interpolation demonstrated. In the context of t-scale computation, a new algorithm for task-independent optimization of edge detection kernels under a given computational constrains and edge characteristics has been developed. An effective ellipse fitting algorithm has been presented that uses both algebraic and geometric distance optimization approaches in a synergistic manner. The improved t-scale computation method has significantly reduced errors (p < 0.01 except for three out of twenty five different combinations of noise and blur) in computed t-scale images as compared to the previous method. A closed form solution for computing interpolation lines using t-scale has been introduced which has contributed to a new t-scale based interpolation algorithm. Experimental results have demonstrated that the new interpolation method outperforms state-of-the-art registration based interpolation techniques on real clinical images and the statistical significance of the improvements was observed.

Acknowledgments

This work is partly supported by internal funds from the Departments of Radiology and Electrical & Computer Engineering at the University of Iowa.

References

- 1. Marr, D. Vision. San Francisco, CA: W. H. Freeman and Company; 1982.
- 2. Witkin, AP. Scale-space filtering. presented at the 8th International Joint Conference Artificial Intelligence; Karlsruhe, West Germany. 1983.
- 3. Koenderink JJ. The structure of images. Biological Cybernatics. 1984; 50:363-370.
- 4. Lindeberg T. Scale-space for discrete signals. IEEE Transactions on Pattern Recognition and Machine Intelligence. 1990; 12:234–254.
- Vincken KL, Koster ASE, Viergever MA. Probabilistic multiscale image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence. 1997; 19:109–120.

- Leung Y, Zhang JS, Xu ZB. Clustering by scale space filtering. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2000; 22:1396–1410.
- Lovell BC, Bradley AP. The multiscale classifier. IEEE Transactions on Pattern Analysis and Machine Intelligence. 1996; 18:124–137.
- Ferraro M, Bocclgnone G, Caell T. On the representation of image structures via scale space entropy conditions. IEEE Transactions on Pattern Analysis and Machine Intelligence. 1999; 21:1190–1203.
- Tabb M, Ahuja N. Multiscale image segmentation by integrated edge and region detection. IEEE Transactions on Image Processing. 1997; 6:642–655. [PubMed: 18282958]
- Pizer SM, Eberly D, Fritsch DS. Zoom-invariant vision of figural shape: the mathematics of core. Computer Vision and Image Understanding. 1998; 69:55–71.
- 11. Elder JH, Zucker SW. Local scale control for edge detection and blur estimation. IEEE Transactions on Pattern Analysis and Machine Intelligence. 1998; 20:699–716.
- Liang, P.; Wang, YF. Local scale controlled anisotropic diffusion with local noise estimate for image smoothing and edge detection. presented at the International Conference in Computer Vision; Bombay, India. 1998.
- Saha PK, Udupa JK, Odhner D. Scale-based fuzzy connected image segmentation: theory, algorithms, and validation. Computer Vision and Image Understanding. 2000; 77:145–174.
- Saha PK, Udupa JK. Scale based image filtering preserving boundary sharpness and fine structures. IEEE Transactions on Medical Imaging. 2001; 20:1140–1155. [PubMed: 11700740]
- 15. Jin, Y.; Laine, AF.; Imielinska, C. SPIE: Medical Imaging. San Diego, CA: 2002. Adaptive speed term based on homogeneity for level-set segmentation; p. 383-390.
- Zhuge, Y.; Udupa, JK.; Saha, PK. SPIE: Medical Imaging. San Diego, CA: 2002. Vectorial scalebased fuzzy connected image segmentation; p. 1476-1487.
- Nyúl, L.; Udupa, JK.; Saha, PK. Task specific comparison of 3D image registration methods. SPIE: Medical Imaging; San Diego, CA. 2001. p. 1588-1598.
- Souza ADA, Udupa JK, Saha PK. Volume rendering in the presence of partial volume effects. IEEE Transactions on Medical Imaging. 2005; 24:223–235. [PubMed: 15707248]
- 19. Saha, PK. Novel theory and methods for tensor scale: a local morphometric parameter. SPIE: Medical Imaging; San Diego, CA. 2003. p. 314-324.
- 20. Saha PK. Tensor scale: a local morphometric parameter with applications to computer vision and image processing. Computer Vision and Image Understanding. 2005; 99:384–413.
- Saha, PK.; Udupa, JK. Tensor scale-based fuzzy connectedness image segmentation. SPIE: Medical Imaging; San Diego, CA. 2003. p. 1580-1590.
- 22. Saha, PK.; Gee, JC.; Xie, Z.; Udupa, JK. Tensor scale-based image registration. SPIE: Medical Imaging; San Diego, CA. 2003. p. 743-753.
- 23. Saha PK, Wehrli FW. A robust method for measuring trabecular bone orientation anisotropy at in vivo resolution using tensor scale. Pattern Recognition. 2004; 37:1935–1944.
- Saha, PK.; Wehrli, FW. In vivo assessment of trabecular bone architecture via three-dimensional tensor scale. SPIE: Medical Imaging; San Diego, CA. 2004. p. 750-760.
- Andalo, FA.; Miranda, PAV.; Torres, RdS; Falcao, AX. A new shape descriptor based on tensor scale. the 8th International Symposium on Mathematical Morphology; Rio de Janeiro, Brazil. 2007. p. 141-152.
- Andalo, FA.; Miranda, PAV.; Torres, RdS; Falcao, AX. Detecting contour saliences using tensor scale. presented at the IEEE International Conference on Image Processing; 2007.
- Grevera GJ, Udupa JK. An objective comparison of 3-D image interpolation methods. IEEE Transactions on Medical Imaging. 1998; 17:642–652. [PubMed: 9845319]
- Thevenaz P, Blu T, Unser M. Interpolation revisited. IEEE Trans Med Imaging. 2000; 19:739–58. [PubMed: 11055789]
- Raya SP, Udupa JK. Shape-based interpolation of multidimensional objects. IEEE Transactions on Medical Imaging. 1990; 9:32–42. [PubMed: 18222748]
- Penney GP, Schnabel JA, Rueckert D, Viergever MA, Niessen WJ. Registration-based interpolation. IEEE Trans Med Imaging. 2004; 23:922–6. [PubMed: 15250644]

- Lee TY, Wang WH. Morphology-based three-dimensional interpolation. IEEE Transactions on Medical Imaging. 2000; 19:711–721. [PubMed: 11055786]
- Sonka, M.; Hlavac, V.; Boyle, R. Image Processing, Analysis, and Machine Vision. 3. Toronto, Canada: Thomson Engineering; 2007.
- 33. Gunn SR. On the discrete representation of the Laplacian of Gaussian. Pattern Recognition. 1999; 32:1463–1472.
- Fitzgibbon A, Pilu M, Fisher RB. Direct least square fitting of ellipses. IEEE Transactions on Pattern Analysis and Machine Intelligence. 1999; 21:476–480.
- 35. Bookstein FL. Fitting conic sections to scattered data. Computer Graphics and Image Processing. 1979; 9:56–71.
- Rosin PL. A note on the least squares fitting of ellipses. Pattern Recognition Letters. 1993; 14:799– 808.
- Gander W, Golub GH, Strebel R. Least-squares fitting of circles and ellipses. BIT Numerical Mathematics. 1994; 34:558–578.
- Penny GP, Schnable JA, Rueckert D, Viergever MA, Niessen WJ. Registration-based interpolation. IEEE Transactions on Medical Imaging. 2004; 23:922–926. [PubMed: 15250644]
- Udupa, JK.; Odhner, D.; Samarasekera, S.; Goncalves, R.; Iyer, K.; Venugopal, K.; Furuie, S. 3DVIEWNIX: an open, transportable, multidimensional, multimodality, multiparametric imaging software system. SPIE: Medical Imaging; San Diego, CA. 1994. p. 58-73.
- 40. Ibanez, L.; Schroeder, W. The ITK Software Guide. Kitware, Inc; 2005.



Figure 1.

A schematic description of t-scale computation. The method starts with edge locations (triangles and black dots) on sample lines emanating from the candidate image point. Following the axial symmetry of an ellipse, the edge points on each pair of radially opposite sample lines are repositioned (black dots to white dots). Finally, t-scale ellipse is computed from repositioned edge points (triangles and white dots).



Figure 2.

Illustration of two connected-intensity profiles. The connected-intensity μ_{UP} successfully eliminates the effect of locally disconnected structures intersecting with the profiles. However, it fails to identify the step-down edges. Therefore another connected-intensity profile μ_{DN} is used to separately locate the step-down edges.



Figure 3.

Errors in a discrete LoG kernel. (a) Sampling of a LoG function and different types of discretization errors after scaling. (b) Relationships among different types of errors with sample interval size Δ .



Figure 4.

Results of geometric distance based optimization of an ellipse (solid line in both (a) and (b)) with initializations using principal component analysis (a: dotted line) and algebraic distance based solution (b: dashed line).



Figure 5.

Results of t-scale computation on a sagittal image slice from an MR brain data. (a) An original image slice with computed t-scale ellipse marked at several locations. (b) HSI color coding used to represent t-scale images. (c,d) Color-coded illustrations of t-scale images computed by the old (c) and the new (d) algorithms. In both (c) and (d), the image on the right is a magnified display of the marked region. Noise and blur induced errors in t-scale image. Reduction of such errors in the result using the new method as compared to the old method is clearly visible.



Figure 6.

Schematic descriptions of the principle of t-scale based image interpolation. (a) Examples of normal vectors at different points inside and outside of a shape. (b) A closed form solution exists for computing the deformation vector using normal vectors.



Figure 7.

Computation of the normal vector from local t-scale. (a) An ambiguity in selecting the

normal vector between \overrightarrow{pb} and $\overrightarrow{pb'}$ the opposite vectors along the minor axis of t-scale. (b) This ambiguity is solved by analyzing t-scale-derived isotropic scales along the direction of the minor axis bb'.



Figure 8.

Intermediate results of t-scale based image interpolation on MR brain data. (a–c) Three successive slices from the BrainWeb MR phantom data. (d,e) Normal vector fields computed from (a,c). (f) Color-coded deformation vector field computed from (d,e) on the slice plane located at the middle of the slice of (a,c). (g) Image interpolated from (a) and (c). Compare the result with (b). (h) Interpolation error computed from (b) and (g). (i) Magnitude of the deformation field.





Figure 9.

(a) Illustration of computing the interpolation line l(t) associated with an image point p on the interpolation plane with the information from t-scale in images of sl_A and sl_B . (b) Same as (a) except that the target point p is close to the lower slice sl_B . Although, the line l'(t) is same in both (a) and (b), the final interpolation line l(t) is changed due to different shifts required under the two cases. Note that the point c_B is close to p_B in (b).



Figure 10.

Illustration of t-scale derived local structure width. (a) A binary phantom image. (b) T-scalederived isotropic scale image. (c) Local structure width map. (d) Computed medialness.



Figure 11.

A schematic description to solve ambiguities of selecting normal vectors at medial points.

At a medial point p_A , there is an ambiguity of the true normal vector between $\overrightarrow{p_A a}$ and $\overrightarrow{p_A a}$. With the assumption that a local deformation is less than half the structure width, the magnitude of the deformation using the correct correspondence of a and b must be less than

that of the wrong correspondence of a' and b. Therefore, \overrightarrow{ab} is selected as the correct correspondence to generate the interpolation line.



Figure 12.

Performance of the old [20] and the new t-scale computation algorithms at varying noise and blurring. (a–c) Original images. (d–e) T-scale images resulting from the old algorithm. (g–i) T-scale images obtained by the new algorithm. Reduction of errors in t-scale computation (spurious dots caused by noise and blur) by the new method as compared to the old one is visible on color illustrations, especially at medium and high noise and blur levels.



Figure 13.

Performance of three interpolation methods on the BrainWeb phantom dataset with additive white Gaussian noise at various levels of signal-to-noise ratio. The mean error was computed as the *MAD* value over the entire image while the standard deviation of errors was computed as the standard deviation of MAD_i values from individual slices. As compared with the two registration based methods, the new method outperforms both of them except for the four cases for which the differences between the B-spline and the new method are statistically identical - marked with "NS" (non-significant).



Figure 14.

Performance of three interpolation methods on the BrainWeb phantom dataset with different multiplicative Gaussian intensity inhomogeneity added (refer to Section 3.2.1). The mean error was computed as the *MAD* value over the entire image while the standard deviation of errors was computed as the standard deviation of MAD_i values from individual slices. MAD_i represents the average interpolation errors on the *i*-th slice for three different inhomogeneity maps at a fixed percentage of inhomogeneity.



Figure 15.

Results of applications of different interpolation methods on the ankle CT data Ankle1. (a– c) Three consecutive image slices from the original data. (d–f) Interpolated results for the central image slice (b) from the two adjacent slices (a,c) using B-spline (d) and Demons (e) based registration methods and the t-scale based method (f). (g–i) Absolute errors by corresponding interpolation method as computed by comparing with the original image slice. Interpolation error is reduced using the t-scale method (p < 0.001).



Figure 16.

Same as Figure 15 but for three consecutive image slices from the pulmonary CT image Lung1 at 0.5 mm slice thickness.



Figure 17.

Same as Figure 15 but for three consecutive image slices from the lower abdominal CT image Abdomen1 at 1mm slice thickness.



Figure 18.

Performance of three interpolation methods on sixteen different medical images selected from different clinical applications. The mean error was computed as the *MAD* value over the entire image while the standard deviation of errors was computed as the standard deviation of *MAD_i* values from individual slices. For all examples, the t-scale based method has outperformed the two registration based methods (p < 0.001).

Table 1

was optimized using 1600 sample lines randomly selected in sixteen different images. The interval size is expressed in terms of the sigma value of a LoG Comparative results of theoretical and experimental optimum interval size for discrete LoG kernels of different lengths. The experimental interval size kernel. The standard deviation of experimental optimum interval size for a discrete LoG kernel separately computed from each individual image is presented on the last row.

			kerr	nel size					
	3	5	7	6	11	13	15	17	19
theoretical	1.5	1.02	0.77	0.64	0.53	0.47	0.41	0.37	0.34
experimental	1.44	0.98	0.78	0.65	0.56	0.50	0.44	0.4	0.36
difference	-0.06	-0.04	0.01	0.01	0.03	0.03	0.03	0.03	0.02
standard dev.	0.09	0.08	0.06	0.05	0.04	0.03	0.03	0.03	0.03

Table 2

Performance of improved t-scale computation algorithm at various levels of noise and blurring. Each row indicates a specific noise level that increases from top to bottom and each column indicates a specific blur level that increases from left to right. Results are reported as mean FOM ± standard deviation.

	B1	B2	B3	B4	B5
N	98.65±0.12	97.92±0.15	97.04 ± 0.26	95.46±0.35	92.36±0.66
N2	98.43±0.11	97.56±0.20	96.54 ± 0.28	94.34 ± 0.46	92.20±0.65
N3	97.95±0.18	96.80±0.28	95.67±0.38	92.82 ± 0.74	91.29±0.76
N_4	96.28 ± 0.45	95.04±0.47	93.68±0.69	91.76 ± 0.68	89.92 ± 0.92
N5	94.43±0.59	93.25±0.78	92.35±0.76	90.44 ± 0.89	89.78 ± 1.06

Table 3

Statistically significant differences are marked by (*) denoting that the new t-scale method outperformed the old approach at the significance level of p < 0Comparison between the interpolation performance of the new and old t-scale computation algorithms at various levels of noise and blurring. Each row indicates a specific noise level that increases from top to bottom and each column indicates a specific blur level that increases from left to right. 0.001. The two methods were equivalent in the three remaining cases.

	B1	B2	B3	B4	BS
Z	0.14(*)	0.16 (*)	0.50 (*)	1.60 (*)	-0.25
N2	0.24 (*)	0.35 (*)	0.63 (*)	1.21 (*)	0.24
N3	0.54~(*)	0.87 (*)	1.15 (*)	-0.20	2.87 (*)
N4	2.35 (*)	2.49 (*)	1.17 (*)	3.09 (*)	3.42 (*)
N5	2.99 (*)	1.75 (*)	1.58 (*)	3.28 (*)	4.61 (*)