Published in final edited form as:
DICTA. 2010 December 1; 2010: 429–434. doi:10.1109/DICTA.2010.79.

An Analytic Approach To Tensor Scale with An Efficient Algorithm and Applications to Image Filtering

Punam K. Saha and
Department of ECE and Radiology, University of Iowa, Iowa City, IA, USA, punam-saha@uiowa.edu

Ziyue Xu
Department of ECE, University of Iowa, Iowa City, IA, USA, ziyue-xu@uiowa.edu

Abstract

Scale is a widely used notion in image analysis that evolved in the form of scale-space theory where the key idea is to represent and analyze an image at various resolutions. Recently, we have introduced a local morphometric scale using an ellipsoidal model that yields a unified representation of structure size, orientation, and anisotropy. In our previous works, tensor scale was described using an algorithmic approach and a precise analytic definition was missing. Here, we formulate an analytic definition for tensor scale in n-dimensional (n-D) images and present an efficient computational solution in 2- and 3-D. Finally, we present an application of tensor scale in medical image filtering. Results of new tensor scale computation algorithm are presented. Also, the performance of tensor scale based image filtering is compared with various approaches of diffusive filtering and the results found are very promising.

Keywords

Scale-space; local scale; tensor scale; image filtering

I. Introduction

Scale [1–3] 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. Scale plays an important role in determining the optimum trade-off between noise smoothing and perception/detection of structures. Also, scale is helpful in breaking a computer vision and image-processing task into a hierarchy of tasks where tasks at higher levels deal with larger structures. Although, scale-space image representations have provided significant insight, it is not obvious – (1) how to unify the information from images at different scales, and (2) how to identify the optimal scale at each individual image point. A knowledge of "local scale" may allow us to spatially tune the neighborhood size in different processes leading to selection of small neighborhoods in regions containing fine detail or near an object boundary, versus large neighborhoods in deep interiors [4]. Also, local scale is useful in developing an effective space-variant parameter controlling strategies [5].

The notion of local morphometric scale was introduced by Saha and Udupa using a spherical model [4, 5] and was applied to different image processing algorithms; see [6] for a survey on local scale. Recently, we have introduced a new local morphometric scale, called "tensor scale" using an ellipsoidal model which provides a unified local parametric representation of structure size, orientation, and anisotropy and its applications have been demonstrated [6–8]. However, previous approaches lack a precise analytic definition for tensor scale and its computational solution in three- or higher-dimensional images is, often, infeasible. In this paper, we introduce a new analytic approach to defining tensor scale in *n*-dimensional (*n*-D) images and present an efficient computational solution using several new algorithms relating gray-scale distance transform and differential geometry. Also, we demonstrate an application of the new tensor scale in image filtering. Weickert *et al*. [9] introduced the notion of structure tensor using a partial differential equation (PDE) Gaussian convolution approaches image intensity and demonstrated its applications in image filtering [9] and adaptive image morphological analysis [10]. Although, structure tensor is a useful concept, it primarily captures information derived from local gradient field and may not directly relate to local structure geometry. For example, in a homogeneous region, structure tensor may not carry meaningful information related to local structure. Here, we formulate tensor scale from a geometric perspective where, at each image point, the tensor captures information related to local structure geometry.

II. Theory and Algorithms

Here, we introduce a new analytic approach to define a local morphometric scale using tensor model. In our previous works, tensor scale was described using an algorithmic approach and its usefulness in several imaging applications have been demonstrated [6–8]. However, a precise analytic definition on tensor scale has been missing and also, previous algorithmic framework is unrealistic for three- and higher-dimensional images due to high computational complexity. In the following, we analytically define tensor scale. Let us consider an image in $\mathbb{R}^n \mid \mathbb{R}$ is the set of real numbers, where multiple objects are defined as partitions by M $(n - 1)$-D manifolds, say, $m_1, m_2, \cdots, m_M$; we refer to these manifolds as *partitioning hyper-surfaces*. Now, let us consider a point $p \in \mathbb{R}^n$ and an ordered sequence of i orthogonal vectors $\boldsymbol{\tau}_1(p), \boldsymbol{\tau}_2(p), \ldots, \boldsymbol{\tau}_i(p)$. An $(n - i)$-D image is formed over the orthogonal complement $W_i^{\perp}$ of the subspace W_i defined by $\boldsymbol{\tau}_1(p), \boldsymbol{\tau}_2(p), \ldots, \boldsymbol{\tau}_i(p)$ with p as the origin where partitioning hyper-surfaces are $(n - i - 1)$ -D manifolds $W_i^{\perp} \cap m_1, W_i^{\perp} \cap m_2, \ldots, W_i^{\perp} \cap m_M$. Let us refer to this image as *orthogonal complement image* of $\boldsymbol{\tau}_1(p), \boldsymbol{\tau}_2(p), \ldots, \boldsymbol{\tau}_i(p)$ at p. *Tensor scale* $\mathbf{T}(p)$ at a point $p \in \mathbb{R}^n$ is defined as an ordered sequence of orthogonal vectors $\boldsymbol{\tau}_1(p), \boldsymbol{\tau}_2(p), \ldots, \boldsymbol{\tau}_n(p)$ inductively defined as follows:

1. $\boldsymbol{\tau}_1(p)$ is the vector from p to the closest point on partitioning hyper-surfaces.
2. Given the first i orthogonal vectors, $\boldsymbol{\tau}_1(p), \boldsymbol{\tau}_2(p), \ldots, \boldsymbol{\tau}_i(p)$, $\boldsymbol{\tau}_{i+1}(p)$ is defined as the vector from p to the closest point on partitioning hyper-surfaces in the orthogonal complement image of $\boldsymbol{\tau}_1(p), \boldsymbol{\tau}_2(p), \ldots, \boldsymbol{\tau}_i(p)$ at p.

We refer to $\boldsymbol{\tau}_1(p)$, $\boldsymbol{\tau}_2(p)$, and $\boldsymbol{\tau}_3(p)$ as *primary*, *secondary*, and *tertiary tensor vectors* of p. The notion of tensor scale defined as above is schematically described in Fig. 1 using a 3-D representation of a rabbit femur bone surface m_1 segmented via μCT imaging; here, cancellous bone and marrow regions are filled in for illustration purpose. As illustrated in the figure, tensor scale $\mathbf{T}(p)$ at a spel p (red dot) in a 3-D image is an ordered sequence $\boldsymbol{\tau}_1(p), \boldsymbol{\tau}_2(p), \boldsymbol{\tau}_3(p)$ of three orthogonal vectors. The first vector $\boldsymbol{\tau}_1(p)$ (red) defines the direction and distance to the closest point on the femur surface. The orthogonal complement plane $W_1^{\perp}$ and the 1-D separating manifold $W_1^{\perp} \cap m_1$ on $W_1^{\perp}$ are shown in the figure; note that the 1-D separating manifold (cyan) is essentially the intersection between the plane $W_1^{\perp}$ (blue) and the partitioning surface m_1 (the femur bone surface) in the 3-D image. The secondary tensor vector $\boldsymbol{\tau}_2(p)$ (yellow) is defined by the point on $W_1^{\perp} \cap m_1$ that is closest to p. Once $\boldsymbol{\tau}_1(p)$ and $\boldsymbol{\tau}_2(p)$ are found, the line (dotted green) on which the tertiary tensor vector $\boldsymbol{\tau}_3(p)$ (green) lie is confirmed; the final direction and the length of $\boldsymbol{\tau}_3(p)$ is defined by finding the closest point on the separating surface along the line. It may be noted that projections of the two dotted lines (yellow and green) on m_1 along the primary tensor vector $\boldsymbol{\tau}_1$ (p) provides two principal directions on m_1 where it meets $\boldsymbol{\tau}_1$ (p); this observation is used in our computational solution in 3-D.

A. Algorithms

In the previous section, we have presented an analytic definition for tensor scale. However, a direct algorithmic realization of the above definition faces two major hurdles – (1) object partitions are not known in real images and (2) computational complexities in three- or higher-dimensions. Here, we present an efficient algorithmic solution for 2- and 3-D images involving edge detection, distance transform, and differential geometric approaches. In an image, we don't know partitioning hyper-surfaces or manifolds used to define tensor scale. However, we may realistically assume that detected edge points in an image lie on these manifolds. Also, because of the fact that these edge points are dense samples on these manifolds, the distance transform from these edge points is a close approximation of distance transform from partitioning manifolds. With this understanding, tensor scale may be computed by using gradient analyses and computational geometric approaches to the distance transform map from edge points in an image.

In this paper, we confine ourselves to 2- and 3-D images, although, the method generalizes to n-D images. We use $\mathbb{Z}^2$ and $\mathbb{Z}^3|\mathbb{Z}$ is the set of integer, to represent a digital space in 2- and 3-D, respectively; we use $\mathbb{Z}^{\mathrm{n}}$ to denote either $\mathbb{Z}^2$ or $\mathbb{Z}^3$. An n-D digital image may be defined by its intensity function f: $\mathbb{Z}^{\mathrm{n}} \to \mathbb{Z}$. Each element of a digital space is referred to as a spel (an abbreviation of "spatial element") whose position is denoted by Cartesian coordinates (x_1, x_2) or (x_1, x_2, x_3), where $x_1, x_2, x_3 \in \mathbb{Z}$. For any two spels $p, q \in \mathbb{Z}^{\mathrm{n}}$, $|p-q|$ denotes the Euclidean distance between the two spels. For any vector $\mathbf{v}$ in $\mathbb{R}^n$, $|\mathbf{v}|$ denotes its magnitude. Let $E \subset \mathbb{R}^n$ denote the set of all edge points in an image; note that a point in E lie on a partitioning manifold of the image. Here, we have adopted an edge detection approach combining both Laplacian of Gaussian (LoG) and Derivative of Gaussian (DoG) operators. Specifically, an edge is located at a zero crossing of LoG if absolute value of its DoG exceeds a threshold. A gray scale distance transform is defined as a function $GDT{:}\mathbb{Z}^n \to \mathbb{R}$, where, $GDT\ (p)|p \in \mathbb{Z}^n$ yields Euclidean distance from p to its closest point in E. As argued earlier, $GDT{:}\mathbb{Z}^n \to \mathbb{R}$ may be considered as an approximate distance transform map from partitioning manifolds. Our tensor scale computation algorithm is primarily based on analyzing this distance transform map.

At a spel p, the gradient of the distance transform map GDT provides the direction to the nearest partitioning hyper-surface, i.e., $\boldsymbol{\tau}_1\ (p)/|\boldsymbol{\tau}_1\ (p)|$ while the magnitude of distance transform determines the distance from the hyper-surface or $|\boldsymbol{\tau}_1\ (p)|$, i.e.,

$$|\boldsymbol{\tau}_1(p)|=GDT(p),$$

and, in 2-D,

$$\boldsymbol{\tau}_1(p)=k\left[\frac{\partial GDT(p)}{\partial x}\ \frac{\partial GDT(p)}{\partial y}\right]^{\mathrm{T}},$$

in 3-D,

$$\boldsymbol{\tau}_1(p)=k\left[\frac{\partial GDT(p)}{\partial x}\ \frac{\partial GDT(p)}{\partial y}\ \frac{\partial GDT(p)}{\partial z}\right]^{\mathrm{T}},$$

where, k is the scalar normalization factor term and it is precisely defined by the magnitude of $\boldsymbol{\tau}_1$ (p) and that of the gradient. Here, we have used the Sobel gradient operator. Once, the primary tensor vector $\boldsymbol{\tau}_1$ (p) is determined, computation of the secondary tensor vector $\boldsymbol{\tau}_2$ (p) in 2-D is straightforward because the vector lies on the straight line L_p perpendicular to $\boldsymbol{\tau}_1$ (p). Thus, $\boldsymbol{\tau}_2$ (p) may be computed by locating the closest separating curve along the straight line L_p. Along a straight line, a partitioning manifold is located when the value of GDT is smaller than '1' and its rate of change along the line is close to '−1'.

Computation of the secondary tensor vector $\boldsymbol{\tau}_2$ (p) is more challenging in 3-D as compared to 2-D. The primary reason behind the difficulty is that, the determination of $\boldsymbol{\tau}_1$ (p) narrows down $\boldsymbol{\tau}_2$ (p) onto a plane P perpendicular to $\boldsymbol{\tau}_1$ (p); however, it may lie along any direction on the plane. This problem may be solved using a differential geometric approach. Specifically, the direction of $\boldsymbol{\tau}_2$ (p) is determined using the principal direction with maximum curvature at the location r where $\boldsymbol{\tau}_1$ (p) meets the nearest separating hyper-surface (see Fig. 1). However, it is not obvious how to compute the principal direction at r because we don't know analytic expressions for separating surfaces; rather, we have discrete sample points (edge locations) on these surfaces. Here, we present a new algorithm to compute the principal direction that works for surfaces represented by discrete sample points. The basic idea here is as follows – first, find the primary tensor vector $\boldsymbol{\tau}_1$ (q) for points in the neighborhood of the candidate spel p. Over a sufficiently small neighborhood, the point of intersection between $\boldsymbol{\tau}_1$ (q) and the separating surface falls in the neighborhood of r (see Fig. 2). More importantly, the angular inclination between $\boldsymbol{\tau}_1$ (q) and the plane P changes more rapidly along the direction of maximum principal curvature and it changes most slowly along the direction of minimum principal curvature. In other words, the projection of the normalized vector $\boldsymbol{\tau}_1$ (q)/|$\boldsymbol{\tau}_1$ (q)| on P takes larger values for qs along the direction of maximum principal curvature and it takes smaller values along the direction of minimum principal curvature. Although our method is primarily based on this theory, to reduce effects of noise and discretization, we compute directions of the two principal curvatures using principal component analysis (PCA) of projections of normalized vector $\boldsymbol{\tau}_1$ (q)/|$\boldsymbol{\tau}_1$ (q)| on P as follows. Let q_1, q_2, …, q_m be m points in the neighborhood of p and let $\boldsymbol{\tau}_1^{'}(q)$ (green vectors on Fig. 2(b,c)) be the projection vector of $\boldsymbol{\tau}_1$ (q)/|$\boldsymbol{\tau}_1$ (q)| onto the plane P. It may be noted in Fig. 2(c) that, primarily, the vectors $\boldsymbol{\tau}_1^{'}(q)$ may fall on one side of the central point. An axial symmetry is imposed to this vector system by adding $-\boldsymbol{\tau}_1^{'}(q)$ (gray) for each original vector $\boldsymbol{\tau}_1^{'}(q)$. Finally, a PCA of all points represented by these vectors is applied to compute the two principal directions; the eigenvector corresponding to larger eigen value gives the direction for maximum principal curvature while the other eigenvector provides the direction for minimum curvature. Projections of these principal directions on the separating surface are illustrated in Fig. 2(d).

B. Application to image filtering

Tensor scale based diffusive filtering is primarily based on the theory of anisotropic diffusion originally proposed by Perona and Malik [11] and subsequently, studied by others [5, 12]. Anisotropic diffusion [11] was originally described to encourage diffusion within a region (characterized by low intensity gradients) while discouraging it across object boundaries (characterized by high intensity gradients). The anisotropic diffusion process at any spel p may be defined as follows:

$$\frac{\partial f}{\partial t}=\text{div }\mathbf{V}=\lim_{\Delta\tau\to 0}\int_{s}\mathbf{V}\cdot \text{d}\mathbf{s},$$

where *f* is image intensity function; *t* is time variable; "div" is divergence operator; **V** = *G***F** is diffusion flow vector; *G* is diffusion conductance function; **F** is intensity gradient vector; Δτ is the volume enclosed by the surface *s* surrounding *p*; and d**s** = **u** d*s* where **u** is a unit vector which is orthogonal and outward-directed with respect to the infinitesimal surface element d*s*. The key idea of anisotropic diffusion [11] is to spatially vary the conductance function as a nonlinear and non-increasing function of gradient magnitude, e.g. $G = \exp(-|\mathbf{F}|^2/2\sigma^2)$ where σ is the controlling parameter.

Although, in conventional diffusive filtering methods [11, 12], diffusion process adapts to local structure orientation using gradient information, the diffusion parameter σ is kept fixed that limits the fine control on and adaptivity to local image structural properties. Weickert *et al.* [9] introduced the notion of structure tensor to control this parameter and demonstrated its use in along-structure smoothing. The motivation of our work is to use geometric tensor information of local structures in filtering that facilitates along-structure smoothing while preserving boundary sharpness. Specifically the controlling parameter σ is determined by local tensor scale in a space- and orientation-variant manner using the ellipsoidal model defined by $\boldsymbol{\tau}_1$ (*p*), $\boldsymbol{\tau}_2$ (*p*), and $\boldsymbol{\tau}_3$ (*p*) as follows:

$$\sigma(p,q)=\chi(\max(\zeta(p,q),\zeta(q,p)))\ \times\ \sigma_{\psi}$$

where, χ is a monotonically non-decreasing function, σ_ψ is the overall noise level in the image, and ζ (*p*, *q*) is the radial length of the ellipsoid Γ (*p*) along *pq* where Γ (*p*) is defined by three tensor vectors $\boldsymbol{\tau}_1$ (*p*), $\boldsymbol{\tau}_2$ (*p*), and $\boldsymbol{\tau}_3$ (*p*) being its semi axes.

III. Results and Discussion

The new tensor scale computation method has been applied on several 2-D and 3-D images. Here, we present the result of application of the method on a 2-D image slice selected from MR brain data available at http://mouldy.bic.mni.mcgill.ca/brainweb/. The image data was downloaded with the following parameters – in plane resolution: 181×217; slice-thickness: 1mm; noise: 3% and intensity non-uniformity: 20%. 2-D tensor scale computation algorithm was applied to an image slice randomly selected from mid-brain region and the result is presented in Fig. 3 (a–c). Results of edge location and gray scale distance transformation are presented in Fig. 3(b). Displaying a tensor scale image may not be straight forward. In 2-D, tensor scale at a spel *p* essentially represents an ellipse Γ (*p*) with its semi axes as $\boldsymbol{\tau}_1$ (*p*) and $\boldsymbol{\tau}_2$ (*p*). We have adopted a color coding scheme to represent 2-D tensor scale where the hue component of color represents orientation of Γ (*p*) and its saturation and intensity denote anisotropy and thickness of Γ (*p*). The color coding scheme at maximum intensity is presented in Fig. 3(d). Results of application of 3-D tensor scale on a pulmonary CT image are presented in Fig. 3(e–g). 3-D tensor scale at a spel *p* essentially represent an ellipsoid Γ (*p*) with three semi-axes $\boldsymbol{\tau}_1$ (*p*), $\boldsymbol{\tau}_2$ (*p*), and $\boldsymbol{\tau}_2$ (*p*). Using three components of color, we can only display an ellipse. Therefore, we display the intersection between Γ (*p*) and display plane which forms an ellipse. Results of both 2-D and 3-D tensor scale computation are visually satisfactory. Using a 2.53 GHz Intel(R) Xeon(R) CPU running under Linux OS, tensor scale computation for a 2D image of size 256×256 takes approximately 1 seconds while it takes approximately 20 minutes for a 3D image of size 256×256×256.

Results of application of different filtering methods are presented in Fig. 4 and Fig. 5. Fig. 4(a) shows a photographic image of a fish and coral reefs. Different 2D diffusive filtering algorithms including the structure tensor based method were applied on the gray scale image and results presented in Fig. 4(b–e). Zoomed images of the rectangular region (marked in (a)) for (a–e) are presented in (f–j), respectively, to illustrate the performance of different methods over regions containing elongated structures. Overall, the method structure tensor based method has outperformed the gradient and curvature based diffusive filtering methods. However, the new tensor scale based method has shown remarkable improvements over all three methods in preserving boundary sharpness of individual structures. None of the methods except the tensor scale based diffusion succeeded to preserve the separation of two branches near the center of red circle.

Fig. 5 (a) shows an original image slice from a pulmonary CT data which was corrupted by white Gaussian noise (Fig. 5(b)). The corrupted 3-D image was subjected to gradient- and curvature-based [13] diffusive filtering and tensor scale based filtering and the results of matching slice are presented in Fig. 5(c–e), respectively. We used the ITK implementation for gradient- and curvature-based diffusive filtering and their recommended values for parameters. As visually appears, both gradient- and curvature-based diffusive filtering algorithms have reduced some noise although, it has blurred some fine structures and also the residual noise is visually apparent. On the other hand, the tensor scale based filtering algorithm has successfully cleaned noise while preserving almost every fine structure visible in Fig. 5(a). To quantitatively evaluate the performance of different methods, we degraded the original 3-D CT data by different levels of Gaussian noise. All three filtering methods were applied on each of these degraded images and the performance in terms of residual noise was computed. Results of this experiment are presented in Table 1. Here, G-, C-, and T-diffusion indicate gradient-, curvature- and tensor scale-based diffusive filtering methods. Original noise was computed by dividing the standard deviation of noise with the mean image intensity. Residual noise was computed as the average absolute pixel-by-pixel difference between filtered and the original image (i.e., prior to adding noise) which was normalized by mean image intensity. Here the original image is considered as the ground truth and the noise within the image is ignored. As may be noted from these results, at all noise levels, the tensor scale-based method has significantly outperformed the other two methods.

In this paper, we have formulated an analytic definition for tensor scale in *n*-D images and have presented an efficient computational solution in 2- and 3-D. Our computational solution is based on new methods for gray scale distance transform and computation of local principal curvature directions on the closest separating hyper-surface represented by discrete edge points. Application of tensor scale in medical image filtering is described and its performance is compared with gradient-based diffusive filtering. Both qualitative and quantitative results have demonstrated major improvements in image filtering using tensor scale as compared to state-of-art methods.

Acknowledgments

This work was supported by the NIH grant R01 AR054439.

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; vol. 50:363–370.

4. Saha PK, Udupa JK, Odhner D. Scale-based fuzzy connected image segmentation: theory, algorithms, and validation. Computer Vision and Image Understanding. 2000; vol. 77:145–174.

5. Saha PK, Udupa JK. Scale based image filtering preserving boundary sharpness and fine structures. IEEE Transactions on Medical Imaging. 2001; vol. 20:1140–1155. [PubMed: 11700740]

6. Saha PK. Tensor scale: a local morphometric parameter with applications to computer vision and image processing. Computer Vision and Image Understanding. 2005; vol. 99:384–413.

7. Andaló FA, Miranda PAV, Torres RdS, Falcão AX. Shape feature extraction and description based on tensor scale. Pattern Recognition. 2010; vol. 43:26–36.

8. Saha PK, Wehrli FW. A robust method for measuring trabecular bone orientation anisotropy at in vivo resolution using tensor scale. Pattern Recognition. 2004; vol. 37:1935–1944.

9. Weickert, J. Anisotropic Diffusion in Image Processing. Stuttgart, Germany: ECMI Series, Teubner-Verlag; 1998.

10. Burgeth, B.; Breuβ, M.; Pizarro, L.; Weickert, J. Scale Space and Variational Methods in Computer Vision. Berlin: Lecture Notes in Computer Science; 2009. PDE-driven adaptive morphology for matrix fields; p. 247-258.

11. Perona P, Malik J. Scale-space and edge detection using anisotropic diffusion. IEEE Transactions on Pattern Analysis and Machine Intelligence. 1990; vol. 12:629–639.

12. Gerig G, Kubler O, Kikinis R, Jolesz FA. Nonlinear anisotropic filtering of MRI data. IEEE Transactions on Medical Imaging. 1992; vol. 11:221–232. [PubMed: 18218376]

13. Yezzi A. Modified curvature motion for image smoothing and enhancement. IEEE Trans Image Processing. 1998; vol. 7:345–352.

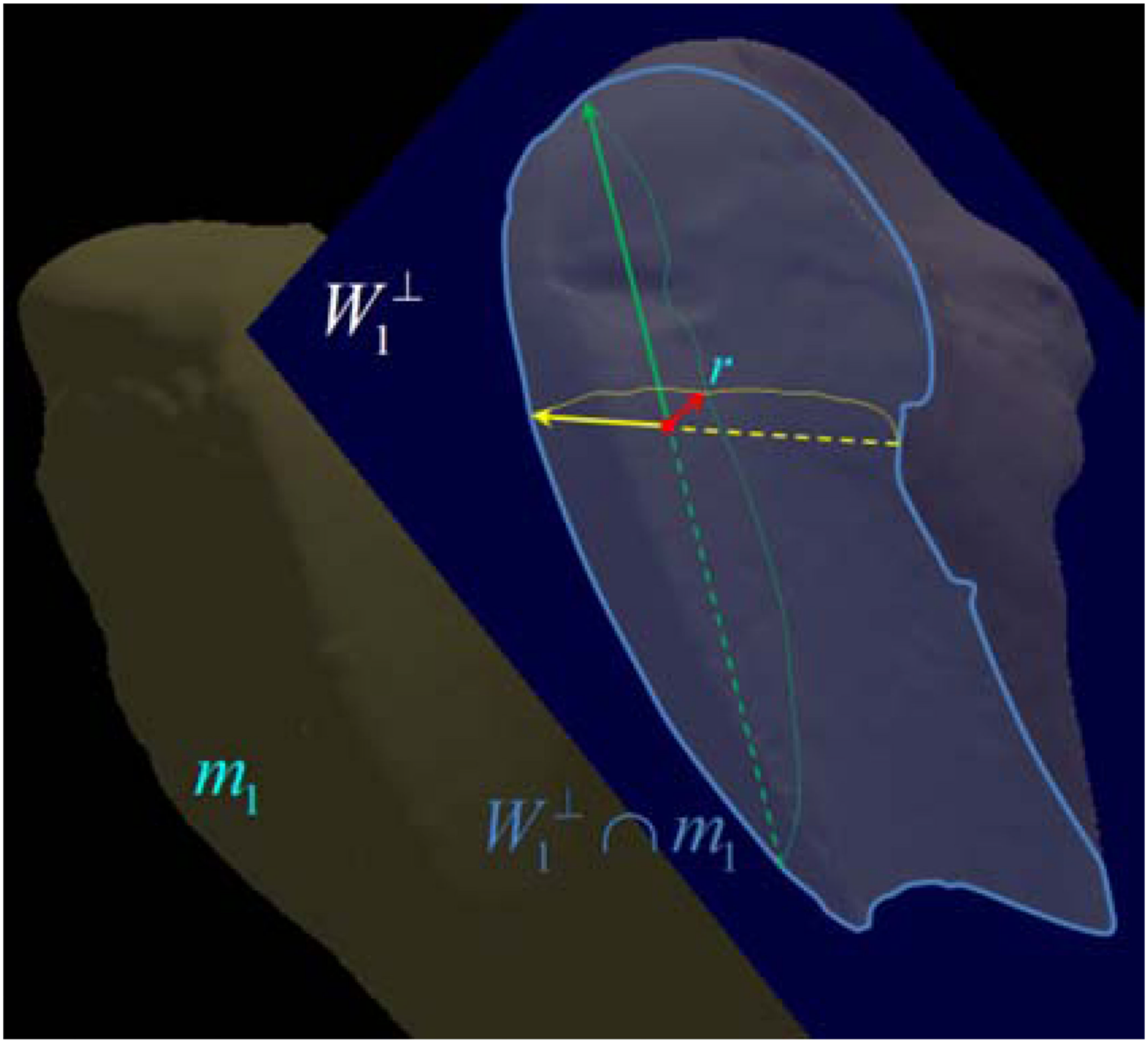

Fig. 1.

An illustration of tensor scale using a rabbit femur bone surface (dark off-white) forming a 2-D manifold m_1. The candidate spel p is shown as a red dot; the point on m_1 closest to p gives the primary tensor vector $\boldsymbol{\tau}_1$ (p) (red). The orthogonal complement plane $W_1^{\perp}$ and the 1-D manifold $W_1^{\perp} \cap m_1$ are shown in blue and cyan, respectively. Secondary tensor vector $\boldsymbol{\tau}_2$ (p) is defined by the point on $W_1^{\perp} \cap m_1$ closest to p; finally, $\boldsymbol{\tau}_3$ (p) is given by the closest point on $W_1^{\perp} \cap m_1$ along the line orthogonal to $\boldsymbol{\tau}_2$ (p). It may be noted that projections of the two lines (dotted yellow and green) on m_1 along $\boldsymbol{\tau}_1$ (p) provide principal directions of m_1 at r, the meeting location with $\boldsymbol{\tau}_1$ (p); this idea is used in our computational solution in 3-D.

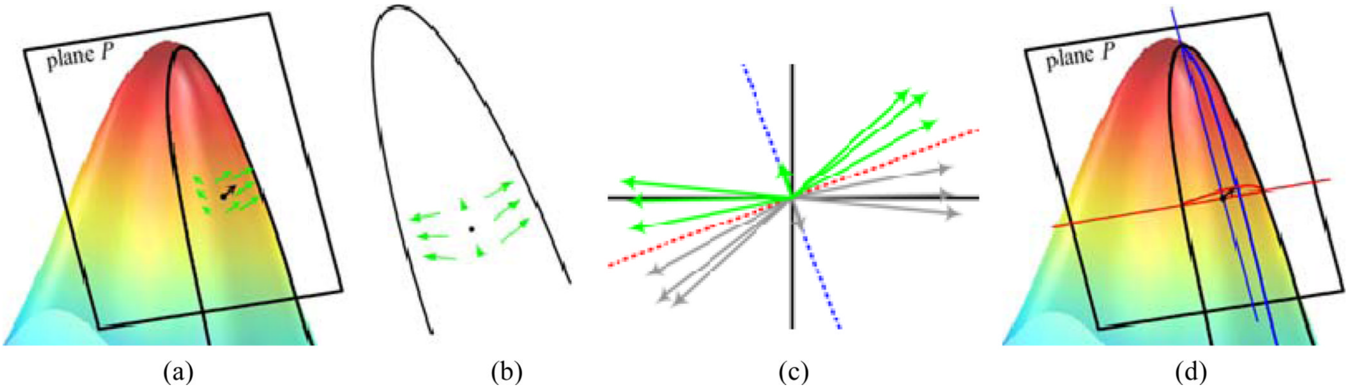

Fig. 2.
Illustration of the computation of secondary tensor vector $\boldsymbol{\tau}_2$ (*p*) in 3-D. (a) A separating surface with primary vector $\boldsymbol{\tau}_1$ (*p*) (black) and $\boldsymbol{\tau}_1$ (*q*) (green) for several *q*s in the neighborhood of *p* (black dot). (b) Projections vectors $\boldsymbol{\tau}_1^{'}(q)$ (green) of normalized vectors $\boldsymbol{\tau}_1$ (*q*)/|$\boldsymbol{\tau}_1$ (*q*)| on *P* along with the curve at the intersection of *P* and the separating surface. (c) Computation of principal directions (dotted lines) using $\boldsymbol{\tau}_1^{'}(q)$ (green) and $-\boldsymbol{\tau}_1^{'}(q)$ (gray), (d) Projection of principal directions onto separating surfaces.

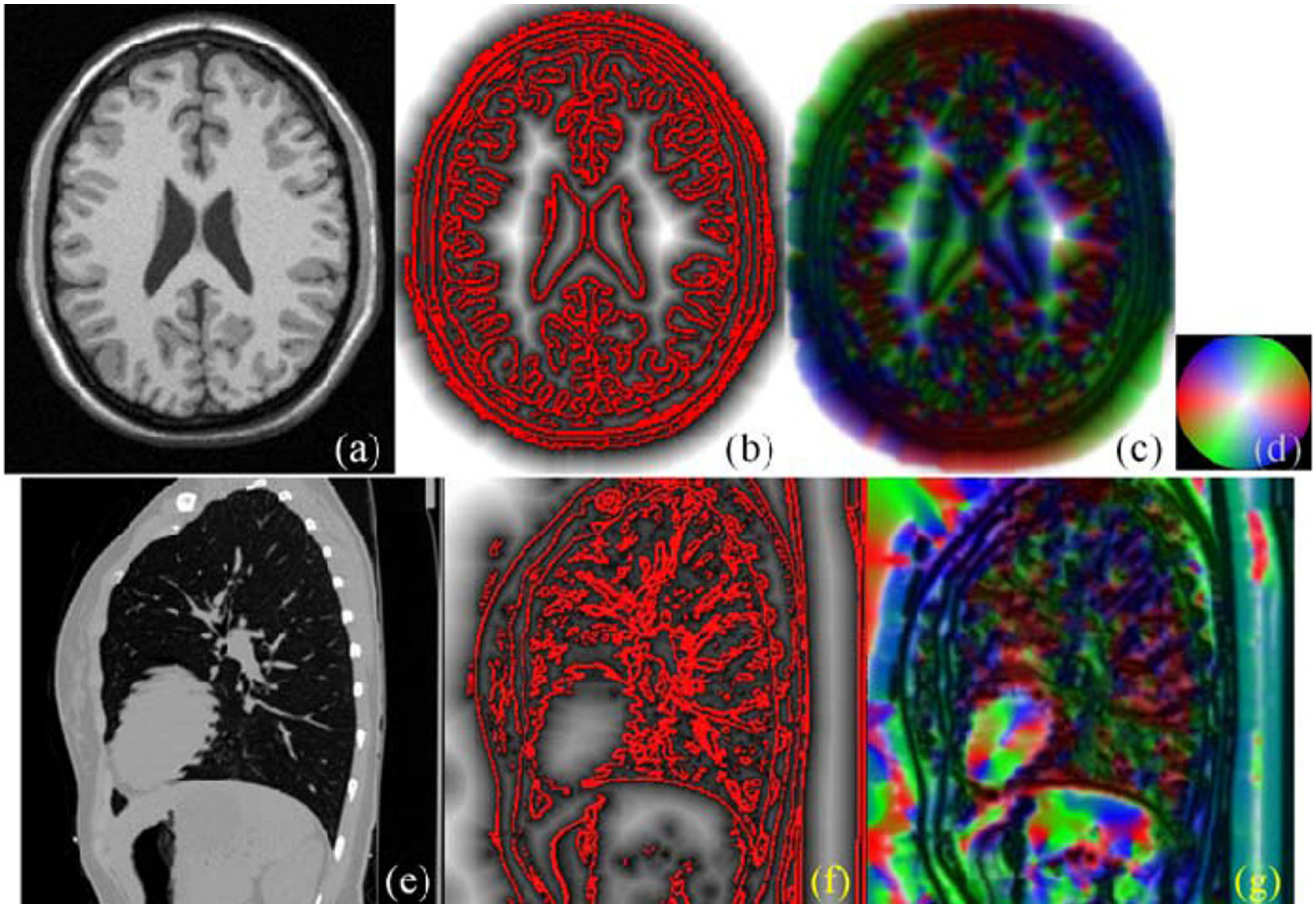

Fig. 3.

Results of tensor scale computation. (a) A 2-D image slice from MR brain data. (b) Computed edge locations (red) and gray scale distance transform. (c) Color coded illustration of tensor scale. (d) Color coding scheme at full intensity. (e–f) Same as (a–c) but for 3-D tensor scale computation. Results are shown on one image slice. See text for explanation.

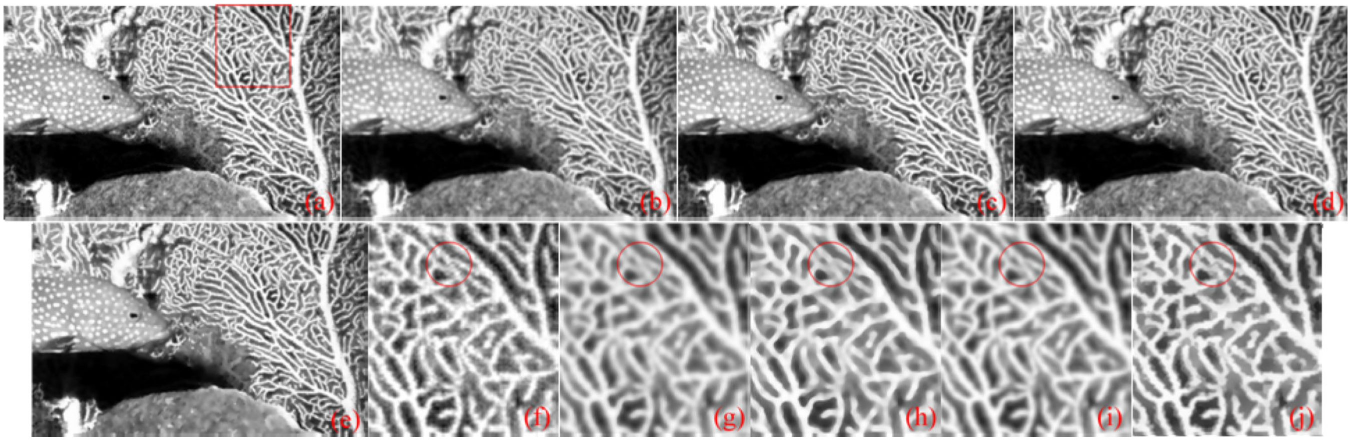

Fig. 4.

Results of 2D image filtering. (a) Original image. (b–e) Results of smoothing using gradient (b), curvature (c), structure tensor (d) and tensor scale (e) based diffusive filtering. (f–j) Zoomed images of the marked region for (a–e) respectively.

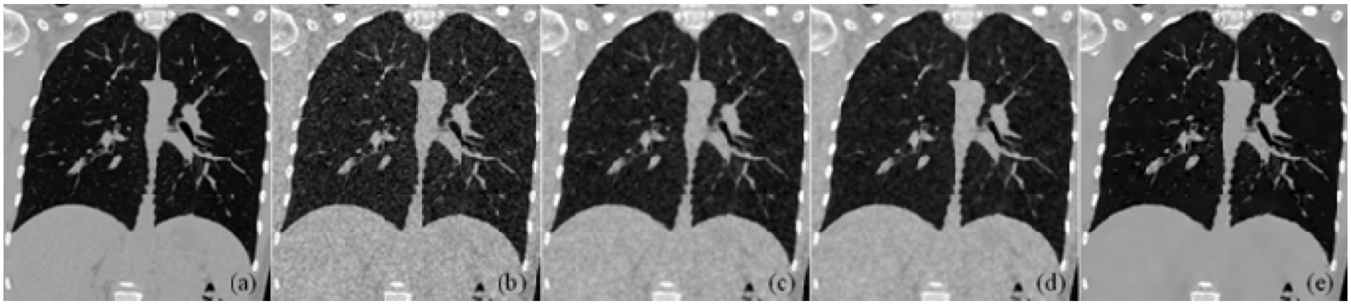

Fig. 5.

Results of image filtering. (a) An original image slice from a pulmonary CT image of a patient. (2) Degraded image after adding Gaussian white noise. (c–e) Results of 3-D image filtering using gradient-based (c), curvature-based (d) and tensor scale-based diffusion (e).

TABLE I

Results of quantitative comparison among three different methods in terms of residual noise after filtering.

Original noise(%)	**Residual noise(%)**		
	G-diffusion	***C-diffusion***	***T-diffusion***
15.0	11.7	9.3	7.0
12.0	9.5	9.0	5.9
10.0	7.7	7.6	5.3
8.0	7.5	6.6	4.4