

First Order Tensor Voting, and Application to 3-D Scale Analysis*

Wai-Shun Tong, Chi-Keung Tang

The Hong Kong University of Science and Technology
Clear Water Bay, Hong Kong
{cstws,cktang}@cs.ust.hk

Gérard Medioni

University of Southern California
Los Angeles, CA 90089-0273
medioni@iris.usc.edu

Abstract

Many computer vision systems depend on reliable detection of 3-D boundaries and regions in order to proceed. In the presence of outliers, missing data, and orientation discontinuities due to occlusion, it is difficult to detect boundaries and interpolate data without over-smoothing important feature curves. In this paper, we address these problems by incorporating first order tensor information into the tensor voting formalism, which is second-order based. To propagate an adaptive smoothness constraint at a preferred orientation non-iteratively, we vote for a first order tensor (or vector) to capture polarity and orientation information. To integrate first and second order tensors, we propose an algorithm for inferring the proper scale based on the continuity constraint, and preserving the finest details. Given a noisy 3-D point set, the new and improved formalism can better localize boundary curves and orientation discontinuities. Unlike many approaches that over-smooth features, or delay the handling of boundaries and discontinuities until model misfit occurs, the interaction of smooth features, boundaries, discontinuities, outliers are encoded at the representation level. We present results from a variety of datasets to show the efficacy of the improved formalism.

1 Introduction

In this paper, we describe two effective and novel contributions to improve the 3-D tensor voting methodology [7]: incorporation of first order tensor information, and automatic detection of scales. These two improvements have a direct impact to the existing formalism, enabling it to solve feature inference and segmentation more effectively and efficiently. While previous effort on second order tensor voting, including a recent one on curvature estimation [14], has reported very good results, the algorithmic issues on first order, and detection of scales in 3-D have remained largely unexplored in literature.

1.1 Motivation and contributions

First, to illustrate the importance of first order information, consider the problem of localizing orientation and depth discontinuities from a noisy depth map. The RENAULT stereo pair is shown in Fig. 1. As usual, the underlying disparity surface is not a closed one. In the original formalism, orientation discontinuity curves can be detected, since normals from two intersecting surfaces in roughly the same proportion give a strong indication

that an intersecting curve exists. However, due to the uneven distribution of tokens around depth discontinuities and occlusion boundaries (Figs. 1 and 2), such detection becomes more difficult, especially in the presence of outliers.

By integrating first order tensors into the formalism, we can now robustly localize depth discontinuities as well. Two views

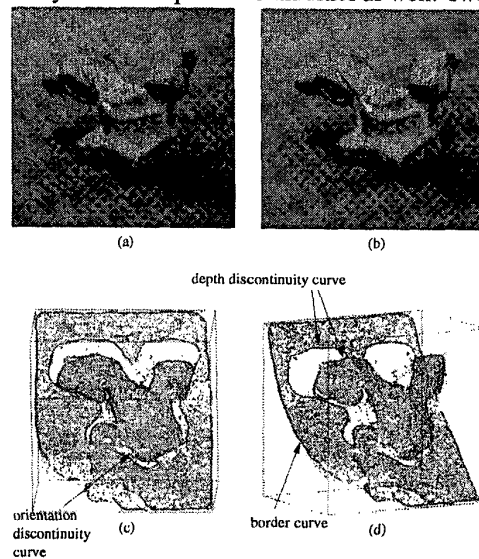


Figure 1 Results on RENAULT.

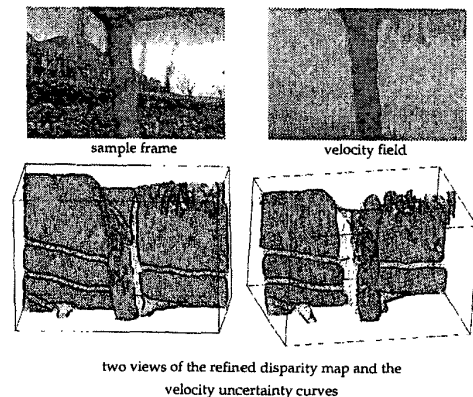


Figure 2 Results on FLOWER GARDEN.

of the feature curves overlaid with the noisy depth data are shown in Fig. 1. Another result is depicted in Fig. 2, in which a 3-frame input is used to produce a velocity field for FLOWER

* This work is supported in part by US National Science Foundation Grant: 9811883, and in part by Research Grant Council of Hong Kong Special Administration Region, China: HKUST6246/00E.

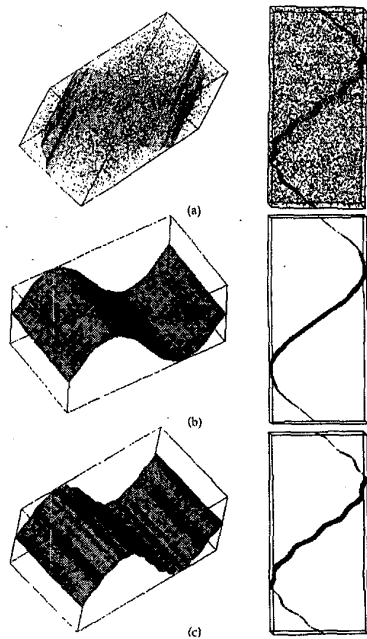


Figure 3 Surface extraction using a large and small scales of analysis.

GARDEN, from which scene depth is derived. Note the uneven distribution of points along the depth discontinuities (e.g., the tree trunk and the flower bed). These two real examples are very difficult, since there are considerable amount of missing data, and that some outlier noise is very close to the correct data. Yet, the qualities of depth and orientation discontinuity curves are very good, and they are automatically detected by our system.

Scale detection constitutes the other contribution of this paper. Illustrated in Fig. 3a is a wavy surface, formed by adding two sine surfaces, one with a large and the other with a small period of oscillation. To make the example more difficult, we add one part of random noise for each part of data in the bounding box. Using a large scale of analysis, we can robustly obtain a very smooth surface, Fig. 3b. Fine details of the surface are revealed by using a smaller scale, Fig. 3c. This observation agrees with the general scale-space theory that features occur at multiple scales.

In this paper, we describe in detail the following technical contributions:

- *First order tensor voting.* We describe and explain the algorithms to perform first order tensor voting for inferring polarity and orientation information in 3-D.
- *Scale inference.* The scale space theory [4] requires the construction of multiscale representation. We integrate first and second order tensor information to derive an algorithm to detect scales of analysis, and apply it in a specific application domain (3-D medical data).

We implemented and tested the improved formalism on real data sets such as 3-D medical images, and noisy range data.

If the data is dense, noise-free, and of sufficient resolution, we can produce quality results comparable to other approaches such as the deformable model, level set, and other visualization

techniques. Otherwise, when one or more of these conditions do not hold, tensor voting degrades more gracefully due to its noise robustness. As shown in the result section, we can work with minimum amount of domain knowledge, and the only necessary assumption we make is an continuity constraint. Our algorithm works with dense data (*McGill Brain* dataset, Fig. 11), sparse data (e.g., [12]), or data with uneven density (*Thorax* dataset, Fig. 10). The last case is made possible by detecting the proper scale. As an added value, we can also work with techniques based on deformable model, since we can give a fairly accurate initial estimate efficiently.

1.2 Related work

Surface and curve inference have been an active research area. Important issues include noise robustness, the presence of orientation and depth discontinuities, and scales of analysis. There is a dilemma in addressing all these issues simultaneously: while a large scale can remove noise, orientation and discontinuities may be smoothed out. A small scale of analysis preserves discontinuities at the expense of noise robustness. Unlike other known approaches (some described below) which compress information into a scalar optimization functional, in this paper, we use a symmetric tensor to encode orientation preference and uncertainty information, and a first order tensor to indicate polarity and discontinuity information. We integrate them to detect features at multiple scales non-iteratively, using a voting algorithm.

Kass, Witkin, and Terzopoulous [5] propose the *deformable surface/model* to fit a surface to a cloud of points. Usually, an initial shape is deformed by iterative techniques to obtain the best parameters. The scale is indicated by a smoothness term in the energy minimization equation. A somewhat careful initialization is needed for “snakes” and similar approaches. The Perona-Malik anisotropic equation [9] deals with scale space and edge detection. In [3], Haralick *et al.* review *mathematical morphology*. It is applied in scale space and many medical image computing applications. Mathematical morphology can be described in the language of set. The dilation and erosion operators, which resembles (but not exactly) the Minkowski sum and subtraction, are used for opening and closing an image. It has nice mathematical properties, but the applicability must be careful [8]. Mathematical morphology is often implemented as a non-linear filtering process, using an iterative technique. The *level set* approach [11] has received much attention in the scale space community. Zero crossings of a higher-dimensional space are calculated, by solving an energy minimization problem. One attraction is that it allows topological changes. Thus, it does not have any limitation on the genus of the reconstruction result, and can deal with non-manifolds. The solution technique, however, is often an iterative minimization solution that requires initialization. Multiresolution techniques and levels of details are also subjects of interest in computer graphics. For example, Hoppe *et al.* [2] use a wavelet-based multiresolution framework to obtain an arbitrary surface by remeshing. *Wavelet* representation [6] has been used to extract surface features at successive scales, by decomposing the original signal using an orthonormal basis. This remeshing can be sensitive to outlier noise. While excel-

lent results are obtained, many computational geometry based techniques are sensitive to outlier noise. The wavelet representation may also be unable to adequately preserve sharp corners and other orientation discontinuities on the underlying surface. In [10], trace points are inferred by estimating the differential structure, which uses an iterative scheme to gather support over a neighborhood. A dense and accurate dataset is needed to obtain a decent estimation of the differentials, especially for second order geometric estimations.

2 Review of second order tensor voting

In this section we review the second order tensor voting [7] to make our presentation complete. In essence, tensor voting makes use of *tensor* for a unified representation, and *voting* for non-iterative data communication. Tensor and voting are brought together by a *voting field*. Voting fields are tensor fields themselves, which postulate the most likely connection by encoding the smoothness constraint. Voting field of any dimensions are derivable from the 2-D *fundamental stick voting field*. We ask the following question: *for a given point P at origin in space, what is the most likely normal (at P) to a curve passing through O and P, and normal to \vec{N} ?* Fig. 4a illustrates the situation. We claim that the *osculating circle* connecting O and P is the most likely one, since it keeps the curvature constant along the hypothesized circular arc. For a detailed theoretical treatment, see [7]. The most likely *normal direction vote* is given by the *normal to the circular arc at P* (thick arrow in Fig. 4a). The length of the normal vector at P, which represents surface saliency or *surfacedness*, is inversely proportional to the arc length OP , and also to the curvature of the underlying circular arc. In spherical coordinates, the decay of the field takes the following form:

$$\overline{DF}(r, \varphi, \sigma) = e^{-\left(\frac{r^2 + \varphi^2}{\sigma^2}\right)} \quad (1)$$

where r is the arc length OP , φ is the curvature, and σ is the size of neighborhood, or scale of analysis, the only free parameter in the formalism. The set of normal direction votes at all P in the 2-D space constitutes the 2-D fundamental stick voting field, Fig. 5, which also shows the tangent version of the field. Each input token *casts votes*, or equivalently is made to align by translation and rotation, with discrete versions of the stick voting field. Each input token gathers votes in its neighborhood whose directions are given by the voting field. The directional votes are aggregated as the *second order* moment collection, or second order symmetric tensor. It is equivalent to an ellipsoid. Its eigensystem is decomposed to give the preferred normal and tangent direction at an input site.

3 First order tensor voting

In our improved formalism, the tensorial representation consists of two parts: a first order tensor and a second order tensor. Intuitively, it can be visualized as an ellipsoid associated with a vector. The major axis of the ellipsoid gives the general direction, and it has two choices of polarities of orientations. The first order tensor completes our representation by specifying the correct orientation. It is essentially a vector, storing the vector sum

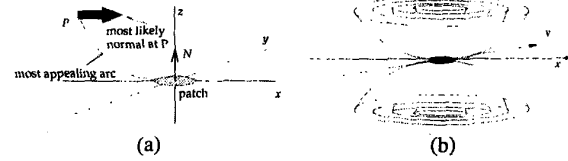


Figure 4 The design of the fundamental voting field. (a) a normal direction vote cast by N is received at P , (b) the 3-D shape of the field.

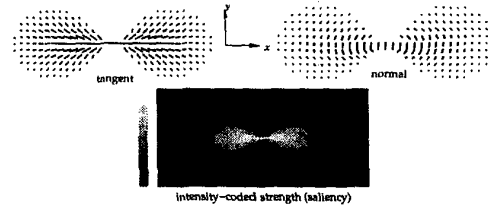


Figure 5 The fundamental 2-D voting field.

of orientations in a neighborhood. Consistent directions with same polarities reinforce each other. Consistent directions with opposite polarities (i.e., \vec{v} and $-\vec{v}$) cancel out each other. Therefore, this first order tensor indicates the degree of agreement in overall orientation.

Depending on the feature (surface or curve) of interest, the first order tensor can indicate *surface endcurviness*, *curve endpointness*, and *region boundariness*. Their mathematical definitions are defined shortly. A salient boundary point has high value(s) of one of these indicators.

By using a first order tensor field, or a vector field, the first order tensor voting algorithms for inferring boundary features are the same as the second order voting (described in appendix A), except that new voting function and vote collection function are defined, which can be plugged into the data communication module directly:

In the integrated formalism, we vote exactly *three* times: 1) second order voting is applied to infer ellipsoids, which encapsulates normal and tangent directions. 2) signs of principal curvatures are estimated [14]. 3) first order tensor voting is applied to infer boundary information, which is the emphasis of the following. The last three rows in the following table summarizes this section.

	Geometric order	Tensorial order	Voting function
junctionness	zero	second	2nd order ball [12]
curviness	first	second	2nd order plate [12]
surfacedness	first	second	2nd order stick [12]
curvatureness	second	second	augmented voting [14]
boundariness	first	first	1st order ball
endpointness	first	first	1st order plate
endcurviness	first	first	1st order stick

3.1 Surface endcurviness

We want to associate each input point Q in 3-D space a 2-tuple (s, \hat{v}) , where s denotes *surface endcurviness*: Q 's likelihood of lying on a surface boundary. \hat{v} is a unit vector indicating the tangent direction to the underlying bounding curve. The curves depicted in Fig. 1 and Fig. 2 are points with large s

values. When all (s, \hat{v}) 's are available, we can trace the curves corresponding to maxima in s along \hat{v} , using a modified curve marching process [12]. At a surface boundary, only one side of a bounding curve is smooth.

As motivated, we only need to change the *voting function* (to propagate the desired feature), and *vote collection* method to produce the first order version. Therefore, it remains to describe the followings:

1. *First order stick voting field.* We obtain a vector field from the normal version of the 2-D *fundamental stick voting field*, by excluding directions/orientations with negative y -coordinates, Fig. 5. This new field thus postulates oriented, smooth connection in the 2-D space, given the orientation $[0 \ 1]^T$ (instead of *unoriented* $\pm[0 \ 1]^T$ as in the original fundamental field). Denote this field by \mathbf{S} . To obtain the 3-D version, we rotate \mathbf{S} about $[0 \ 1]^T$, integrating the contributions by vector addition: $\int_{\alpha} \mathbf{S} d\alpha$. We use this voting field to encode and enforce the smoothness constraint. The 3-D shape of this first order field is similar to Fig. 4b, except that only one stack of the “bowls,” instead of two as shown.
2. *Vote direction, strength, and collection.* Refer to Fig. 6a–b. Suppose a point P with normal \vec{N}_p casts a vote \vec{N} to Q , using the first order stick voting field by calling GEN-NORMALVOTE (appendix A). We compute the curvature direction \hat{v}_p , obtained by intersecting Π_{PQ} (the plane at P that contains Q and \vec{N}_p ; note by definition of the 2-D stick field they all lies on the same plane.) with the tangent plane \mathbf{T}_q at Q , fig. 6c. As shown by the thick arrow on \mathbf{T}_q , only one orientation of the curvature direction is chosen. Hence, \hat{v}_p is given by: $\vec{N}_q \times ((P - Q) \times \vec{N}_q)$.

The scalar endcurveness s , which indicates the likelihood, is inversely proportional to the arc length PQ , and related to the magnitude of the inner product $\vec{N} \cdot \vec{N}_q$ (i.e., whether P and Q can be connected *smoothly* by the first order voting field). s_p thus indicates the strength of the vote.

In COMBINE (appendix A), instead of computing the second order moment, we collect first order moment, or vector sum, by $s\hat{v} = \sum_p s_p \hat{v}_p$ for all votes (s_p, \hat{v}_p) received in Q 's neighborhood. The 2-tuple (s, \hat{v}) of Q is thus obtained. A large $|s|$ indicates high endcurveness.

3.2 Curve endpointness

We associate each point Q another 2-tuple (s, \hat{v}) , where s denotes *curve endpointness*: Q 's likelihood of being a curve endpoint. \hat{v} is a unit tangent vector pointing toward the interior of a curve. At a curve endpoint, only one side of its tangent is smooth. Again, we have:

1. *First order plate voting field.* To encode curve smoothness, we rotate and integrate \mathbf{S} by vector addition: $\int_{\beta} \int_{\alpha} \mathbf{S} d\alpha d\beta$. The resulting 3-D field is a plate-like vector field. It describes a curve whose tangent is normal to this plate, similar to [7], but the field is now a first order one. We use this field to enforce curve smoothness.
2. *Vote direction, strength, and collection.* Refer to Fig. 7. Suppose a point Q with tangent \vec{t}_q receives a plate vote \vec{t}_p

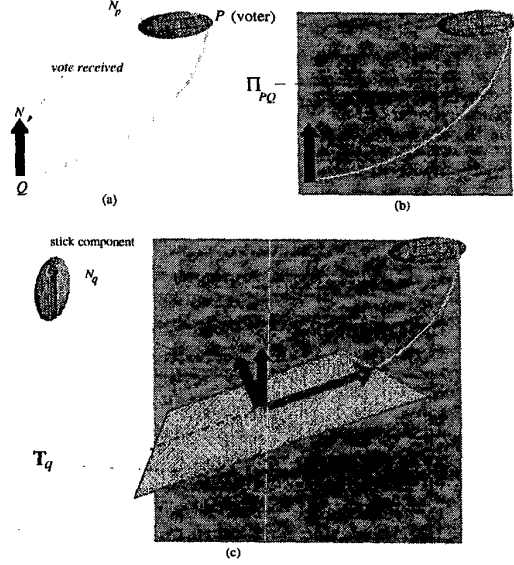


Figure 6 (a) Voter P casts a normal direction vote \vec{N} at Q , (b) the plane at P that contains Q and \vec{N}_p , (c) the direction of the resulting vote is indicated by the red arrow, with saliency depending on the magnitude of \vec{N} and the angle between \vec{N} and \vec{N}_q .

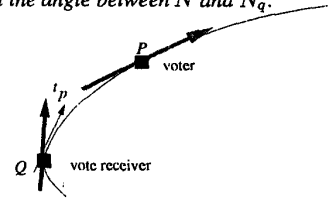


Figure 7 Illustration of curve endpoint vote definition. from P . The vote strength s_p is obtained by projecting \vec{t}_p onto \vec{t}_q , i.e., $s_p = \frac{\vec{t}_p \cdot \vec{t}_q}{\|\vec{t}_p\| \|\vec{t}_q\|}$. Let \hat{v}_p be $\frac{\vec{t}_q}{\|\vec{t}_q\|}$. Thus, we have $s\hat{v} = \sum_p s_p \hat{v}_p$. (s, \hat{v}) is thus obtained. Large $|s|$ corresponds to high likelihood of a curve endpoint.

3.3 Region boundariness

Similarly, a 2-tuple (s, \hat{v}) is associated with an input point Q . s indicates *region boundariness*, Fig. 8: Q 's likelihood of lying on a bounding surface in 3-D. \hat{v} indicates is a normal orientation pointing toward the “inside” of the bounding surface.

1. *First order ball voting field.* To propagate the uniformity constraint, we use the isotropic first order ball voting field, obtained by rotating and integrating \mathbf{S} in three principal directions: $\int_{\gamma} \int_{\beta} \int_{\alpha} \mathbf{S} d\alpha d\beta d\gamma$.
2. *Vote direction, strength, and collection.* Refer to fig. 8. The vote direction and strength are directly obtained from the first order ball voting field. Votes are collected using vector addition (i.e., use the COMBINE we modified in section 3.1).

4 Scale detection

In this section, we present an algorithm that uses first and second order tensor voting to adjust the scales of analysis. The scale space theory [4] states that the construction of multiscale representations is necessary if no prior information is available on

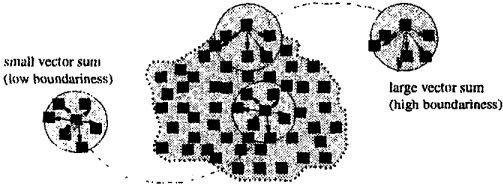


Figure 8 Illustration of voting for boundary. Small vector sum indicates low boundariness, as the neighborhood tokens cast vote at all directions. Large vector sum indicate high boundariness since vote are received at a specific set of directions only.

the appropriate scales. Therefore, the desired solution depends on domain knowledge or criterion. In medical image analysis, for instance, we can assume the absence of salient junction: although the corresponding surfaces of organs and tissues can sometimes be very convoluted, they should still be smooth everywhere.

We first use a 2-D synthetic image to illustrate the process. By thresholding image intensity, we produce an initial set of tokens in Fig. 9a. First, we obtain a set of polarity directions indicating region boundariness, Fig. 9b. Then, the polarities are refined into normal directions. A high junction saliency indicates highly convoluted features, or errors resulting from missing data (both cases are characterized by a set of inconsistent normals violating the continuity constant). Note that we also detect signs of Gaussian curvature (by [14]). A neighborhood with high junction saliency is removed, Fig. 9c. We then perform endpoint detection, Fig. 9d. The scale is then progressively increased, until the detected endpoints are connected smoothly, or the largest permissible scale has been reached, Fig. 9e. The same voting process is applied to produce a curve, Fig. 9f.

The 3-D version assumes no domain knowledge other than smoothness is depicted in the following:

```

SurfaceExtract( $\sigma_1, \sigma_n$ )
1. Threshold intensity (Fig. 9a)
2. Vote for bounding surface polarities using  $\sigma_1$  (Fig. 9b)
3. for each neighb ortho o(b) do {
   $\sigma \leftarrow \sigma_1$ 
  repeat {
    Vote for boundary surface normals using  $\sigma$  (Fig. 9c)
    Vote for sign of curvatures using  $\sigma$ 
    if ( $\max(\text{neighb ortho o(b)}. \text{junctionness}) > \mu_j$ ) (fig. 9c)
      Remove tokens around b's neighborhood
    Vote for endcurve using  $\sigma$  (fig. 9d)
    if ( $\max(\text{neighb ortho o(b)}. \text{endcurveness}) > \mu_c$ )
       $\sigma \leftarrow \sigma + k$ 
    Vote for surface with curvature-based voting fields [14]
    (Fig. 9e-f)
  } until (neighb ortho o(b).endcurveness <  $\mu_c$ ) or ( $\sigma > \sigma_n$ )
}

```

In the above, σ_1 and σ_n are the smallest and largest scales respectively. They indicate an approximate range, and are easy to find since they need not be very accurate. In step 3 above, we want to find the smallest $\sigma \in [\sigma_1, \sigma_n]$, such that continuity is satisfied at the smallest possible scale, and that finest details are preserved. In each iteration, smoothness and/or continuity is detected by the two conditional statements in the algorithm. We have exactly four cases:

1. *High junctionness and high endcurveness.* The smoothness constraint is violated at the current scale σ , and continuity cannot be achieved. We increase σ by some integer constant k to enforce continuity in a larger scale in the next step.
2. *High junctionness and low endcurveness.* The smoothness constraint is violated at the current scale σ . Normally, the removal of a small neighborhood should automatically result in high endcurveness, if the junctionness is due to highly convoluted features. However, if the high junctionness is due to error resulting from missing data, salient endcurves cannot be detected.
3. *Low junctionness and high endcurveness.* Continuity is violated at the current scale. We increase the scale of analysis, similar to case 1.
4. *Low junctionness and low endcurveness.* This means the current scale of analysis is optimal for maintaining both smoothness and continuity. Therefore, the repeat loop stops if this condition is true.

Typical value of k is 1 in a $150 \times 150 \times 150$ grid for medical data. Experiments in [7] show that the step size k is not critical, since tensor voting is applicable over a reasonable range in $[\sigma_1, \sigma_n]$.

5 Results

We run experiments on difficult, real and simulated medical data. All the actual running times are measured on a Pentium III 600 MHz, with 384MB RAM. Fig. 10 shows the reconstruction result of the *thorax* dataset. This is a real set of CT scans (courtesy of University of Washington Health Science Center). Only 12 slices are used, whose z -coordinates span unevenly from 110–180. After intensity thresholding, we perform 2-D first order voting with the ball voting field on the 12 slices to infer the region boundaries. The segmentation results on three sample slices are shown in fig. 10a. Next, we stack together the resulting thorax boundaries into a 3-D volume of dimensions $170 \times 114 \times 70$, Fig. 10b. To make the example more difficult, we add a total of 12000 noisy points to the volume (i.e. signal:noise = 1:2). Second order tensor voting is then performed to refine the boundary polarities into normals, and to detect the sign of curvatures. The surface can be extracted by another pass of tensor voting using a larger scale σ . Two views of the segmented surface description are shown in fig. 10c. We do not use any assumption other than the continuity constraint in our reconstruction. The main processing times are tabulated in table 1.

	σ	Processing times (secs)	Remarks
Intensity thresholding	n/a	10	
First order voting	3	45	2-D region boundary
Second order voting	15	495	3-D normal sign of curvatures surface extraction
total		550	

Table 1 Processing times for the thorax dataset.

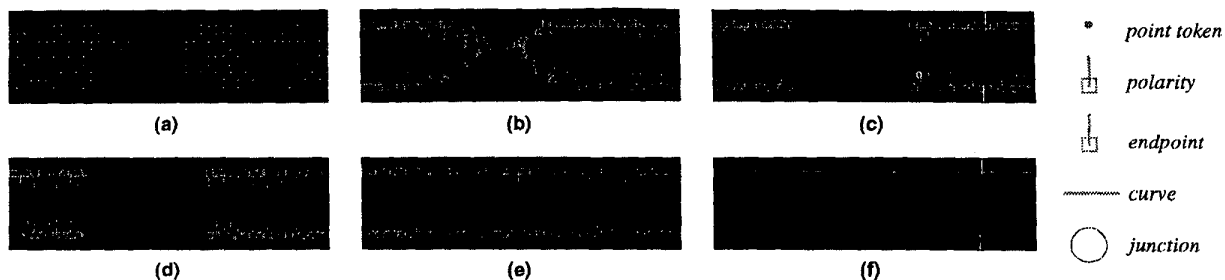


Figure 9 Illustration of the integrated formalism (a) initial set of tokens, (b) polarities; a junction is labeled, (c) removal of tokens around the detected junction, (d) endpoint detection, (e) missing tokens are inferred during the multiscale analysis, (f) final curve extraction result.

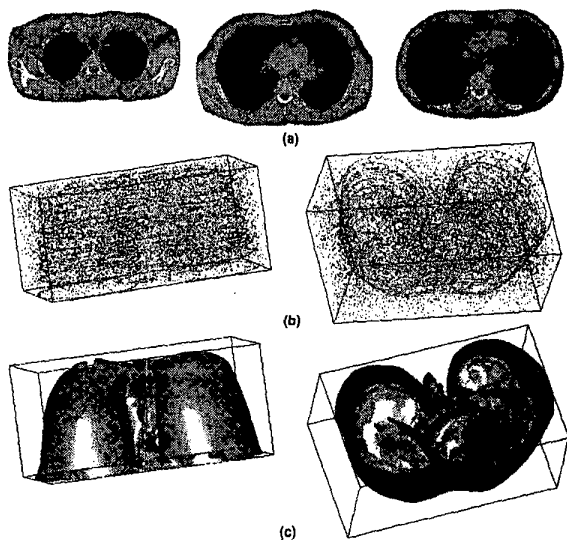


Figure 10 Results on the 12 CT scan of the Thorax dataset. (a) 3 slices of the CT scans showing the inferred boundaries of the thorax (the red curves), (b) 3-D boundary points before surface extraction, (c) result of the surface extraction using our method.

The *McGill Brain*, from the Brainweb [1], is used to evaluate our algorithm. The dimensions of the MRI dataset are $181 \times 217 \times 181$, with resolution at $1mm^3$. We are interested in three types of tissues: the cerebrospinal fluid or CSF, the gray matter, and the white matter. This is a very challenging dataset. All surfaces are very convoluted. Without any *a priori* assumption or initialization, we can still robustly separate two closely spaced surfaces that define the respective boundaries of the CSF ribbon (or *CSF/GM*), the gray matter ribbon (or *GM*, in green), and white matter (*GM/WM*), Figs 11a-c, by using first order voting for 3-D boundary detection. To further delineate the two bounding surfaces of the gray matter, we perform two optional intersections, based on domain knowledge that CSF encloses GM, which in turn encloses WM. Thus, $CSF/GM \cap GM$ gives the outer cortical surface, whereas $GM/WM \cap GM$ produces the inner cortical surface. Fig. 11d shows the resulting intersections: the red curves indicate the outer surface, and the blue curves indicate the inner surface. Note the good match we have obtained. Fig. 11e-f show the automatically generated surface models for the inner and outer cortical surface, respectively. Table 2 summarizes the main processing times of *McGill Brain*.

	σ			Processing times (secs)		
	CSF	GM	WM	CSF	GM	WM
Thresholding	n/a	n/a	n/a	35	41	43
First order	0.5	0.5	0.5	468	615	723
Second order	2	2	2	≈ 1500	≈ 1500	≈ 1500

Table 2 Processing times for the McGill Brain dataset.

We perform error analysis on the resulting gray matter surfaces obtained above (Fig. 11), for which ground truth is available. Qualitatively, the results are very satisfactory. Our quantitative comparison is as follows: the true bounding surfaces of gray matter are first derived from the set of pixels known to be gray matter (ground truth) by manual expert tracing. We count the number of pixels classified as cortex surfaces, and check against the true boundary pixels. Let *TP%* be the true positive percentage that indicates our correct classification. Let *FP%* and *FN%* be the false positive and negative percentage respectively. We obtained the following results: $TP = 89.76\%$, $FP = 4.36\%$, $FN = 5.10\%$.

6 Conclusion

In this paper, we study first order tensor voting, and describe an algorithm that integrates first and second order tensors to detect scales of analysis. This improved formalism integrates orientation preference and uncertainty at the representation level. No over-smoothing or delayed decision based on model misfit is used, since orientation, outliers, and discontinuities are inferred simultaneously through a data communication or voting step. By itself, the 3-D first order tensor voting can be used as a plugin, which produces reliable 3-D edges and curves. For example, surface endcurves localize the orientation and depth discontinuities; 3-D region boundaries give the bounding surface for a point cluster. In this paper, we make use of this algorithm to detect feature curves in real and noisy stereo, motion, and to infer segmented surface descriptions for 3-D medical data. We show the efficacy of the approach by making a worst possible case: no domain knowledge is used, and that a lot of outlier noise is present. Our encouraging results show that the formalism improves the tensor voting algorithm, both on the theoretical and on the practical levels.

A Algorithms for 3-D tensor voting

We detail the general tensor voting algorithm [13] in this section. The voter makes use of GENTENSORVOTE to cast a tensor

vote to vote receiver (votee). Normal direction votes generated by GENNORMALVOTE are accumulated using COMBINE. An 3×3 *outTensor* is the output. The votee thus receives a set of *outTensor* from voters within its neighborhood. The resulting tensor matrices can be summed up by ADDTENSOR (not shown here), which performs ordinary 3×3 matrix addition. This resulting matrix is equivalent to an ellipsoid.

Algorithm 1 GENTENSORVOTE (voter, votee)

It uses GENNORMALVOTE to compute the most likely normal direction vote at the votee. Then, plate and ball tensors are computed, by integrating the resulting normal votes cast by voter. They are used to vote for curves and junctions.

```

for all  $0 \leq i, j < 3$ , outTensor[i][j]  $\leftarrow$  0
for all  $0 \leq i < 2$ ,
    voterSaliency[i]  $\leftarrow$  voter[ $\lambda_i$ ] - voter[ $\lambda_{i+1}$ ]
voterSaliency[2]  $\leftarrow$  voter[ $\lambda_2$ ]
if (voterSaliency[0] > 0) then
    vecVote  $\leftarrow$  GENNORMALVOTE (voter, votee)
    {Compute stick component }
    COMBINE (outTensor, vecVote)
end if
transformVoter  $\leftarrow$  voter

for  $i = 1$  to 2 do
    // 1: plate, 2: ballif (voterSaliency[i] > 0) then
    // count[i] is a sufficient number of samples uniformly
    // distributed on a unit ( $i + 1$ )-D sphere.
    while (count[i]  $\neq$  0) do
        transformVoter[direction]  $\leftarrow$  random[direction]  $\leftarrow$  GEN-
        RANDOMUNIFORMPT()
        if ( $i \neq 2$ ) then
            /* Compute the alignment matrix, except the isotropic ball
            tensor */
            transformVoter[direction]  $\leftarrow$  voter[eigenvectorMatrix]  $\times$ 
            random[direction]
        end if
        vecVote  $\leftarrow$  GENNORMALVOTE (transformVoter, votee)
        COMBINE (outTensor, vecVote, voterSaliency[i])
        count[i]  $\leftarrow$  count[i] - 1
    end while
end if
end for
return outTensor

```

References

[1] *BrainWeb*, McConnel Brain Imaging Center at the Montreal Neurological institute (<http://www.bic.mni.mcgill.ca/>)

[2] M. Eck, T. DeRose, T. Duchamp, H. Hoppe, T. Lounsbery, W. Stuetzle, "Multiresolution analysis of arbitrary meshes," *Computer Graphics (SIGGRAPH 1995 Proc.)*, pp. 173-182, 1995.

[3] R.M. Haralick, C. Lin, J.S.J. Lee, and X. Zhuang, "Multi-Resolution Morphology," in *Proc. Intl. Conf. Comput. Vision*, 1987.

[4] T. Lindeberg. Scale-space representation: Definition and basic ideas. In *CVonline: On-Line Compendium of Computer Vision*. R. Fisher (ed). Available: "<http://www.dai.ed.ac.uk/CVonline/>".

[5] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: An Active Contour Models," *Int'l. J. Computer Vision*, 1(4):321-331, 1987.

Algorithm 2 GENNORMALVOTE (voter, votee)

A vote (vector) on the most likely normal direction is returned.

```

v  $\leftarrow$  votee[position] - voter[position]
/* voter and votee are connected by high curvature? */
if (angle(voter[direction], v) <  $\pi/4$ ) then
    return ZeroVector {smoothness constraint violated}
end if
/* voter and votee on a straight line, or voter and votee are the same
point */
if (angle(voter[direction], v) =  $\pi/2$ ) or (voter = votee) then
    return voter[direction]
end if
Compute center and radius of two osculating hemisphere as shown
in Fig. 4b - change to one in section 3.1
/* assign stick vote */
stickvote[direction]  $\leftarrow$  center - voter[position]
stickvote[length]  $\leftarrow$   $e^{\frac{\sigma^2 + \epsilon \rho^2}{\sigma^2}}$  {equation (1)}
stickvote[position]  $\leftarrow$  votee[position]
return stickvote

```

Algorithm 3 COMBINE (tensorvote, stickvote, weight)

It performs tensor addition, given a stick vote.

```

for all  $i, j$  such that  $0 \leq i, j < 3$  do
    tensorvote[i][j]  $\leftarrow$  tensorvote[i][j] + weight  $\times$  stickvote[i]  $\times$ 
    stickvote[j]
end for

```

[6] S. G. Mallat, "A Theory for Multiresolution Signal Decomposition: the Wavelet Representation," *IEEE Trans. Patt. Analysis Machine Intell.*, vol. 11, no. 7, pp. 674-693, 1989.

[7] G. Medioni, M.-S. Lee, C.-K. Tang, *A Computational Framework for Segmentation and Grouping*, Elsevier Science, 2000.

[8] K.-R. Park and C.-N. Lee, "Scale-Space Using Mathematical Morphology," *IEEE Trans. Patt. Analysis Machine Intell.*, vol. 18, no. 11, 1996.

[9] P. Perona and J. Malik, "Scale-space and Edge Detection Using Anisotropic Diffusion," *IEEE Trans. Patter Analysis Machine Intell.*, vol. 12, no. 7, pp. 629-639, 1990.

[10] P.T. Sander and S.W. Zucker, "Inferring Surface Trace and Differential Structure from 3-D Images," *IEEE Trans. Patt. Analysis Machine Intell.*, vol. 12, no. 9, pp. 833-854, 1990.

[11] J.A. Sethian, *Level Sets Methods*, Cambridge University Press, 1996.

[12] C.-K. Tang and G. Medioni, "Inferring Integrated Surface, Curve, and Junction Description from Sparse 3-D Data Sets," *IEEE Trans. Patt. Analysis and Machine Intell.*, vol. 20, no. 11, pp.1206-1223, 1998.

[13] C.-K. Tang, G. Medioni, M.-S. Lee, "N-dimensional Tensor Voting, and Application to Epipolar Geometry Estimation," *IEEE Trans. Patt. Analysis and Machine Intell.*, accepted.

[14] C.-K. Tang and G. Medioni, "Robust Estimation of Curvature Information from Noisy 3-D Data for Shape Description," in *Proc. IEEE Intl. Conf. Comput. Vision* (Corfu, Greece), pp.426-433, 1999.

[15] The National Library of Medicine's *Visible Human Project*. http://www.nlm.nih.gov/research/visible/visible_human.html

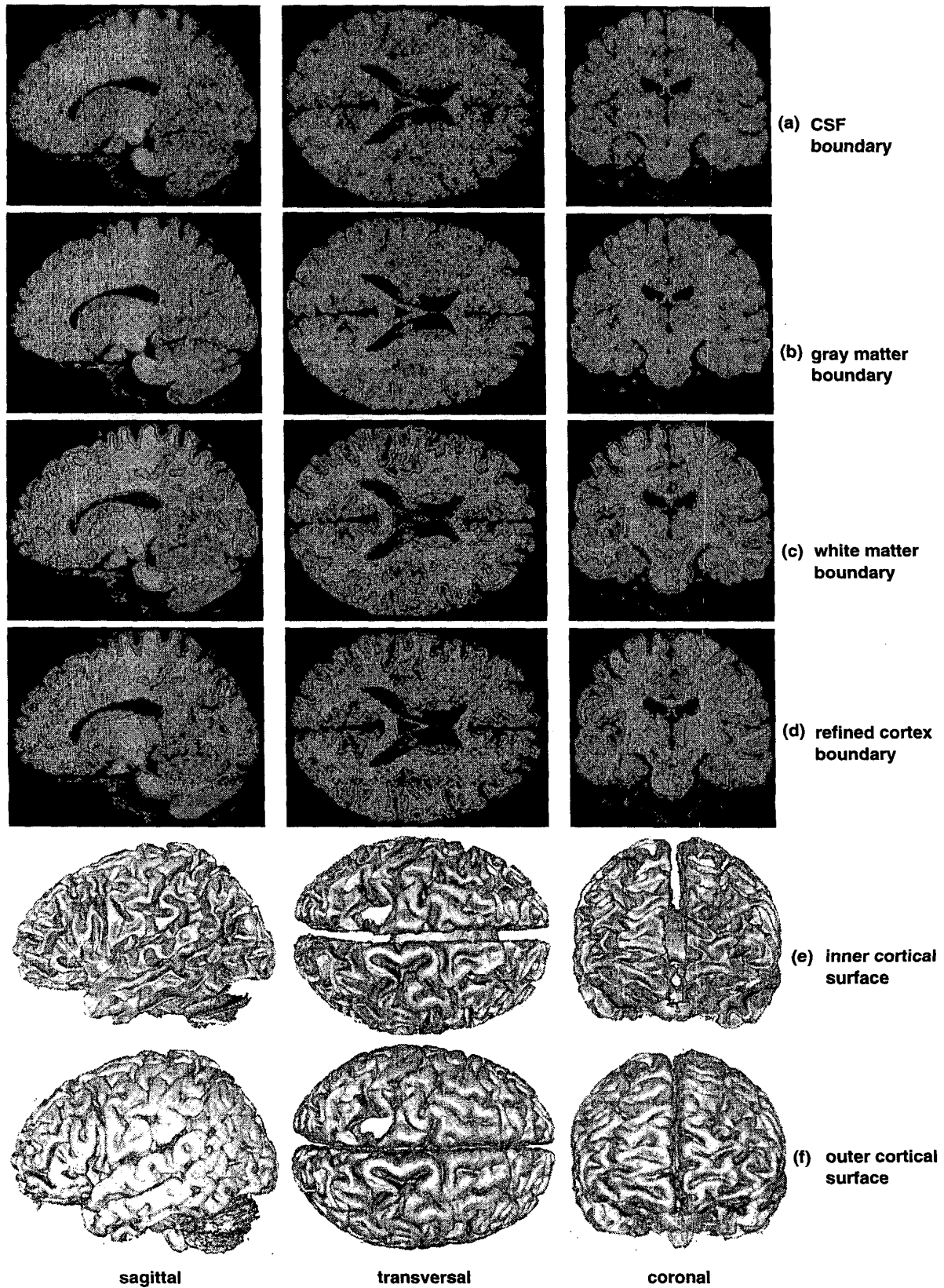


Figure 11 Results on the McGill Brain dataset. The sagittal, transversal, coronal views of the extracted (a) CSF, (b) gray matter, and (c) white matter boundaries are shown. The inner cortical surface (shown in blue), and the outer cortical surface (shown in red) are shown together in (d). In (e) and (f), three views of the inner and outer cortical surface mesh are shown.