

5 Recovering Surfaces, 3-D Intersections, and 3-D Junctions Using Perceptual Constraints

Gideon Guy and Gérard Medioni

We treat the problem of recovering surfaces from an incomplete set of input measurements, by applying perceptual constraints to the data. This extends our 2D perceptual Grouping work [64] to three dimensions. We show how both the Extension fields and the Saliency indicators, which were used for the 2-D case, can be elegantly generalized to 3-D.

We treat sparse “clouds” of non-oriented points, oriented points, and partial surfaces, in a uniform and *non-iterative* way. We are able to handle scenes of any genus, any number of discontinuities, and of any number of objects, without a priori knowledge or special considerations. The result is in the form of three dense saliency maps for surfaces, intersections between surfaces, and 3-D junctions. These saliency maps can then be used to guide a “following” process to generate a CAD model of surfaces, space curves, and 3-D junctions. We present some preliminary results on computer-generated images.

5.1 Introduction

Perceptual organization has gained popularity in the Computer Vision research in the past few years, and its importance has been widely recognized. First proposed by Lowe [67], and later by numerous researchers (e.g. [59,60,69], and see [63] for a more complete review), perceptual considerations have been used for a variety of problems in Computer Vision. All of above attempts took as input two dimensional image features.

Among the perceptual constraints used, the most common are: Co-Linearity, Proximity, Simplicity, and Co-Curvilinearity. These constraints are used to handle gaps and errors in input data, and assist in a higher-level description. The same kind of task is present in 3-D inputs, where some 3-D data is available, but it is not complete and/or it is noisy. Such input data is normally acquired by range imaging or as a result of a process that finds depth from X (stereo, shape from shading etc.). Here the task is to describe the underlying surfaces.

Much work has been done in fitting surfaces to clouds of points. The deformable models approach (first proposed by Kass *et al.* [65] for 2D, and in [70] for 3D) attempts to deform an initial shape so that it fits a set of points through energy minimization. The deformable shape is subject to certain constraints (such as smoothness, stiffness

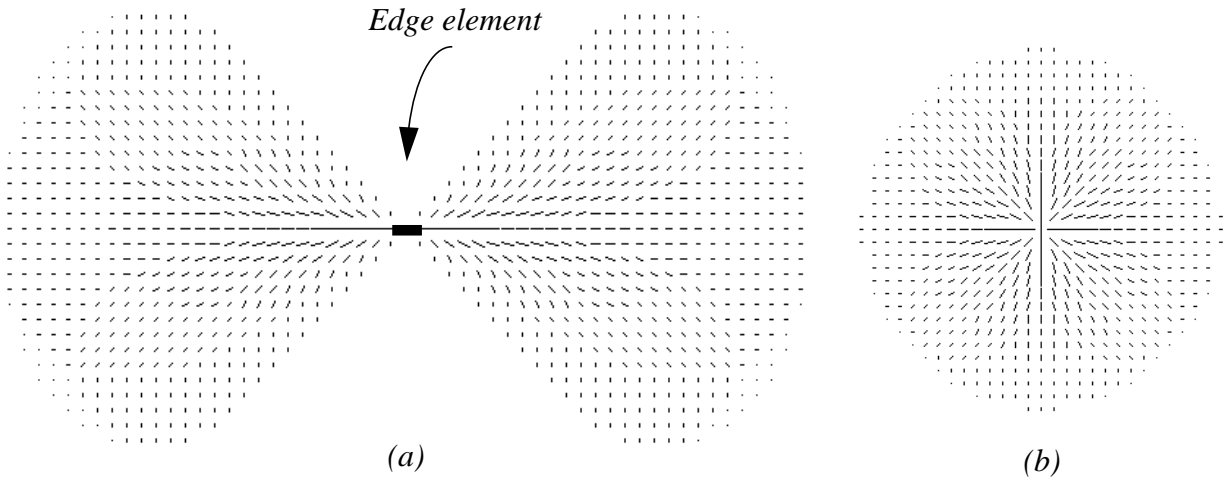


Figure 5.1 (a) The 2-D Extension field (b)the 2-D point field.

etc.) and the point data set behaves as an attractive energy field. Sander [61] have recently proposed a local algorithm to describe surfaces from a set of points. A quadratic surface is estimated around each data point, and an iterative process refines the local surfaces, and creates graph-like connections between compatible patches. Others have used similar methods (for a summary see [66]).

All of the above methods are computationally expensive as an iterative process takes place. Also in many of the methods, only one genus-zero object can be described at any one time, and surface boundaries and discontinuities are not always easy to describe.

We start by briefly discussing our older 2-D work, emphasizing the derivation and justification of the combination mechanisms. This will serve as an introduction to the derivation of the 3-D combination mechanisms, which share the same paradigms.

5.2 From 2D fields to 3D fields

In our 2-D work [64] we describe two basic fields, namely, the Extension field and the point field. The Extension field is used when edgels are present in the input, and the point field is used whenever non-oriented features are present. The 2-D version of the two fields are shown in Figure 5.1. The elements of the 2-D Extension field describe the most likely orientation of a curve passing everywhere in space. The above field is used as a mask in a so-called directional convolution, in which the mask is oriented along segments of the input image. votes that consist of strength and orientation are accumulated at each site of the image, to later determine the saliency of such site. The result is thus a saliency map, where high values denote high likelihood of a curve passing there. The actual combination at each site is described next and paves the way to the 3D case.

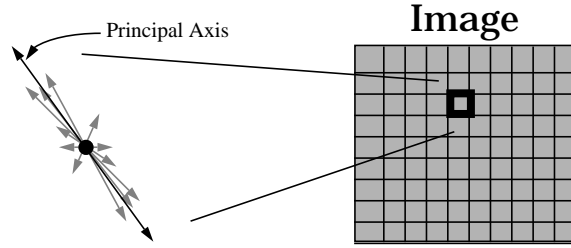


Figure 5.2 The principal axis of the votes collected at a site is taken as an approximation of the preferred direction.

5.2.1 Combination per site - The 2-D case

Ideally, we would want an averaged majority vote regarding the preferred orientation at a given position. In practice, we treat the contributions to a site as being vector weights, and compute moments of the resulting system. Such a physical model behaves in the desired way, giving both the preferred direction and some measure of the agreement. We use the direction of the principal axis ($EVmin$) of that physical model as the chosen orientation (See equation (5.1)).

$$\begin{bmatrix} m_{20} & m_{11} \\ m_{11} & m_{02} \end{bmatrix} = \begin{bmatrix} EVmin \\ EVmax \end{bmatrix} \begin{bmatrix} \lambda_{min} & 0 \\ 0 & \lambda_{max} \end{bmatrix} \begin{bmatrix} EVmin^T & EVmax^T \end{bmatrix} \quad (5.1)$$

This acts as an approximation to the desired majority vote, without the need to consider the individual votes.

The saliency map *strength* values are taken as the values of the corresponding λ_{max} at each site. So, large values would indicate that a curve is likely to pass through this point. This map can be further enhanced (as shown in the next section) by considering the eccentricity, or $1 - (\lambda_{min}/\lambda_{max})$. When that value is multiplied by the previous saliency map we achieve better selectivity, and only *curves* are highlighted. This results in a map defined by $\lambda_{max} - \lambda_{min}$.

5.2.1.1 Justification - The 2-D case

Basically, what we are looking for is a function that takes positive vectors as input and results in a measure of the agreement in their orientation. The result should satisfy several criteria:

- We want the result to be normalizable, so that we can compare different sites on a standard scale.
- The measure needs to be monotonically increasing with the addition of positive contributions.
- It should give higher values to 'better' (more directed) spatial arrangements of vectors.
- We want the effect of proximity to be independent of the affect of agreement.

It is easy to show how the model behaves when a single vector is added to it. Assume the variance-covariance matrix is as follows at state t :

$$C^t = \begin{bmatrix} m_{20}^t & m_{11}^t \\ m_{11}^t & m_{02}^t \end{bmatrix} \quad (5.2)$$

The sum of the eigenvalues is the trace of the matrix:

$$\lambda_{\min}^t + \lambda_{\max}^t = m_{20}^t + m_{02}^t \quad (5.3)$$

Now adding a new vector $V = [R\cos\theta, R\sin\theta]^T$ to the system will result in a new state $t+1$:

$$\lambda_{\min}^{t+1} + \lambda_{\max}^{t+1} = m_{20}^t + m_{02}^t + (R\cos\theta)^2 + (R\sin\theta)^2 = m_{20}^t + m_{02}^t + R^2 \quad (5.4)$$

Note that the angle θ has disappeared on the r.h.s. of (5.4). This means that the sum of eigenvalues is independent of the orientations of the voting vectors and can hence be used as an indicator of proximity (a wider sense of proximity of course), and as a primitive saliency measure.

Equation (5.4) can obviously be written as:

$$\lambda_{\min} + \lambda_{\max} = \sum_{i=1}^N R_i^2 \quad (5.5)$$

Where N is the number of segments in the original image.

We define the eccentricity $E = 1 - \lambda_{\min}/\lambda_{\max}$ as a measure of agreement. Obviously this value is between 0 and 1¹. Our intuitive notion of ‘agreement’, or of a majority vote on a continuous scale, is consistent with the above definition. This means that in all cases where we feel that collection A has better ‘agreement’ than collection B , the corresponding eccentricity values will share the same relationship (i.e. $E(A) > E(B)$). This is not to say that both functions are equal, but merely that both are monotonic.

Eccentricity values by themselves cannot perform as saliency measures since sites with very little voting strength can produce high eccentricity values. In fact, consider a site far away from where the ‘action’ is, which accepts exactly *one* vote (This can happen in practice). The eccentricity value is 1, but the site is of no importance.

1. Since $\lambda_{\min} \leq \lambda_{\max}$ and are both non-negative for a semi-positive definite matrix.

However, Consider λ_{\max} itself. Obviously,

$$\frac{\lambda_{\min} + \lambda_{\max}}{2} \leq \lambda_{\max} \leq \lambda_{\min} + \lambda_{\max} \quad (5.6)$$

By (5.6) it is bounded from both sides by the proximity measure in (5.5) *and* has the eccentricity coded into it: When the value leans towards the left side of (5.6), eccentricity is low and vice-versa.

Thus, λ_{\max} is chosen as the raw saliency measure in our scheme.

This choice however, may still amplify locations which are very strong in terms of number of votes, but weak in eccentricity². The product of E and λ_{\max} produces the desired result, termed the *enhanced saliency* measure SM, or:

$$SM = \lambda_{\max} \cdot (1 - \lambda_{\min}/\lambda_{\max}) = \lambda_{\max} - \lambda_{\min} \quad (5.7)$$

Thus, $\lambda_{\max}-\lambda_{\min}$ is chosen as the enhanced saliency measure.

It is important to note that other functions of the eigenvalues can also satisfy the same conditions of monotonicity, but the ones chosen seem to be the *simplest* possible indicators of the desired behavior.

5.2.1.2 Detection of Junctions

A junction is defined as a *salient* point which also has a *low* eccentricity value.

Regular (non-junction) points along a curve are expected to have high eccentricity values. On the other hand, junction points are expected to have low eccentricity, since votes are accumulated from several different directions. By combining the eccentricity and the eigenvalue at a point, we acquire a continuous measure of the likelihood of that site being a junction. We redefine our previous definition of eccentricity slightly, so that low eccentricity scores high, or:

$$(E = \lambda_{\min}/\lambda_{\max}) \Rightarrow 0 \leq E \leq 1 \quad (5.8)$$

The product of our new eccentricity measure and the raw saliency measure λ_{\max} yields the junction saliency operator:

$$E \cdot \lambda_{\max} = (\lambda_{\min}/\lambda_{\max}) \cdot \lambda_{\max} = \lambda_{\min} \quad (5.9)$$

This process creates a *Junction Saliency map*. Interestingly enough, this map evaluates to just λ_{\min} at every site (as shown in Equation (5.9)), which simply means that the *largest non-eccentric* sites are good candidates for junctions. By finding all *local maxima* of the junction map we localize junctions.

2. For example, accumulation points and junctions! (where $\lambda_{\min} \equiv \lambda_{\max}$)

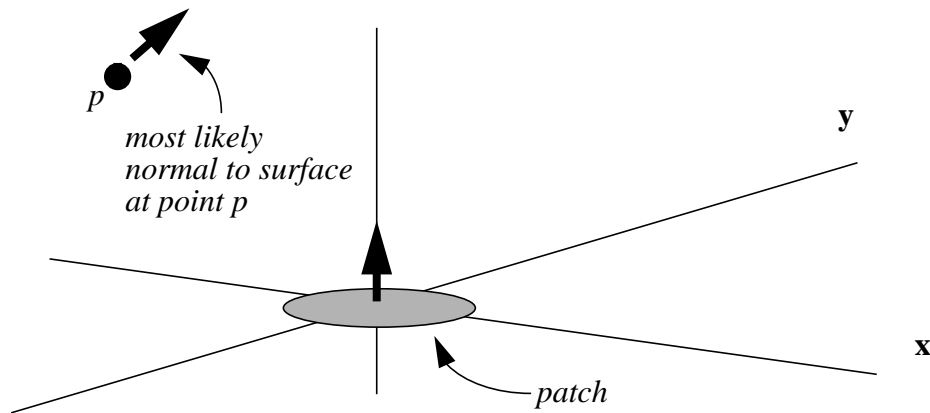


Figure 5.3 What is the most natural normal to a surface passing through point p and at the same time tangent to the patch at the origin?

5.2.2 3-D fields

In the 3-D case we would like to treat three elementary features, namely, a patch, a curve segment, and a point in space. A patch has a known 3-D normal, a curve segment has all possible normals lying on a plane, and a point in space has absolutely no directional data. We will construct a separate field for each of these features.

5.2.2.1 The construction of the Patch Extension Field

We assume that a patch with a known normal is available, and we ask the following question: for a given point in space, what is the most likely normal to a surface passing through that given point and also tangent to the original patch? Figure 5.3 illustrates that issue. It is clear that the desired normal at point p can be found by looking at a 2-D scenario, where both the origin and point p are on a plane. This reduces the problem to a 2D one, where the 2-D Extension field can be applied. Thus, constructing the 3-D Extension field is merely revolving the 2D Extension field around its vertical axis. This is illustrated in Figure 5.4 . Note that unlike the 2-D field, where each field element pointed *in* the direction of the most likely curve, in the 3-D case, each vector points in the direction of the *normal* at that location. This makes later stages of computation much simpler.

5.2.2.2 The curve segment Extension Field

Here we deal with a primitive with partial information regarding the orientation of a surface passing through it. All we know is that the given segment has to lie on the desired surface. Again we ask the question: What is the most likely surface to pass through a point p in space and have the segment at the origin lying on it? The answer is very simple. A segment and a point in space define exactly one plane³. And since a

3. Except, of course, the points co-linear with the segment.



Figure 5.4 The general shape of the 3-D Extension Field. The lower part is omitted from the sketch, but is similar to the upper part. Field elements are normal to the surfaces shown, and were also removed for display purposes.

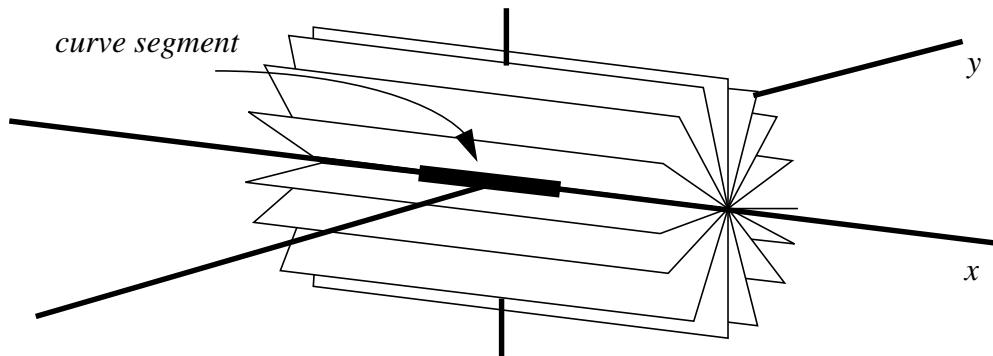


Figure 5.5 The general shape of the curve segment field. All planes go to infinity, with diminishing strength. The field elements are in reality normals to the drawn planes

plane is the best surface in terms of the perceptual constraints, it is also the most likely to appear.

A practical way of constructing this field is to take the Patch Extension field and convolve it with a multi-directional patch⁴. This last operation is similar to revolving the patch Extension field around itself along the x (or y) axis (referring to notation in Figure 5.4). By symmetry considerations, it is simple to show that the resulting field will have the correct orientations everywhere in space (as shown in Figure 5.5). This construction also determines the strength values at every site of the field.

5.2.2.3 The 3-D Point field

The 3-D point field is even simpler to derive. The only thing in common to all surfaces passing through a point in space and the origin, is that the line connecting the

4. The 2-D point field was constructed by the exact same way from the 2-D Extension field (in [63]).

two points is on all of them. However, in this case, there does not exist a single maximum likelihood normal. That is, at each point in space, *many* normals are equally likely. Luckily, they all lie on a plane perpendicular to the line formed by the point in question and the origin. We thus choose to describe the contribution of all these normals with a single 3-D vector pointing in the direction of the above line. We will later show how such a voting vector is treated to determine saliencies.

5.2.3 Directional Convolution

The process of computing the saliency maps is similar to the 2-D case. We will describe it here again for sake of completeness. Computing the Saliency maps can be thought of as a directional convolution with one of the above fields (mask). The resulting map is then a function of a collection of fields, each oriented along a corresponding short normal in 3-D. The whole operation is performed in a 3-D grid or array. Each site accumulates the ‘votes’ for its own preferred orientation and strength from every other site in the image. These values are combined at a site as described next.

When the input data consists of non-oriented features (e.g. 3-D points), a 2-pass convolution was found to work best, first applying the point field in order to estimate orientations, and then the patch Extension field, for the final results. The same procedure is performed if curve primitives are present in the input image.

5.2.4 Combination at each Site

Combination per site is really the process of choosing, for each site, the preferred normal that will show up in the final saliency map. The 3-D case will be derived based on the same methodology used for the 2-D scheme (as described earlier in this paper).

5.2.4.1 The 3-D case

Here we need to consider a 3x3 variance-covariance matrix, as shown in Equation (5.10), where λ_{max} , λ_{mid} , and λ_{min} signify the three sorted eigenvalues of the system. (Note that the 3-D discussion assumes that normals to the desired surfaces are doing the actual voting⁵!)

$$\begin{bmatrix} m_{200} & m_{110} & m_{101} \\ m_{110} & m_{020} & m_{011} \\ m_{101} & m_{011} & m_{002} \end{bmatrix} = \begin{bmatrix} EV_{min} \\ EV_{mid} \\ EV_{max} \end{bmatrix} \begin{bmatrix} \lambda_{min} & 0 & 0 \\ 0 & \lambda_{mid} & 0 \\ 0 & 0 & \lambda_{max} \end{bmatrix} \begin{bmatrix} EV_{min}^T & EV_{mid}^T & EV_{max}^T \end{bmatrix} \quad (5.10)$$

The three eigenvectors will correspond to the three principal directions of an ellipsoid in 3-D, while the eigenvalues describe the strengths and agreement measures of the 3-D votes.

5. Except for the non-oriented case, which is discussed in a separate section.

As before λ_{\max} is bounded on both sides by the sum of eigenvalues (which corresponds to raw strength) and at the same time encodes the eccentricity. When it leans toward the right hand side of Equation (5.11), eccentricity is high and when it leans toward the left hand side, eccentricity is low.

$$\frac{\lambda_{\min} + \lambda_{\text{mid}} + \lambda_{\max}}{3} \leq \lambda_{\max} \leq \lambda_{\min} + \lambda_{\text{mid}} + \lambda_{\max} \quad (5.11)$$

Thus, λ_{\max} is selected as a raw saliency measure for surface normals, and the corresponding eigenvector determines the orientation of that normal.

To further enhance the measure we can require that the other two eigenvalues be low compared to the λ_{\max} . This can be achieved by looking at the difference, $\lambda_{\max} - \lambda_{\text{mid}}$. The expression will yield high values only when both λ_{mid} and λ_{\min} are small. The most likely normal to the surface, is merely the eigenvector corresponding to λ_{\max} .

The same logic holds for intersections between surfaces. Here, we would like to look at λ_{mid} as a saliency measure. When it is high, so must λ_{\max} , and the location is really characterized by votes coming from exactly two separate surfaces.

Thus, λ_{mid} is chosen as a raw saliency measure for intersection between surfaces.

Again, this measure can be enhanced by considering $\lambda_{\text{mid}} - \lambda_{\min}$. This last expression will exclude locations along intersection curves that belong to a higher-level intersection (i.e. a junction). The direction of the curve is given by the eigenvector perpendicular to the two surfaces, or the one corresponding to λ_{\min} .

Lastly, we claim that large values of λ_{\min} will correspond to locations where three (or more) smooth surfaces intersect, or a 3-D junction. It is clear that if λ_{\min} is large, so are the other two. Three large eigenvalues describe a spherical distribution of normals, meaning normals from many different orientations have voted for that point in space.

λ_{\min} is thus chosen as the junction saliency map.

The 2-D and the 3-D results can be summarized in Table 6:. The highlighted col-

Table 6: 2-D and 3-D results

Feature	2-D raw saliency	2-D enhanced saliency	3-D raw saliency	3-D enhanced saliency
Junction	λ_{\min}	λ_{\min}	λ_{\min}	λ_{\min}
curve	λ_{\max}	$\lambda_{\max} - \lambda_{\min}$	λ_{mid}	$\lambda_{\text{mid}} - \lambda_{\min}$
surface			λ_{\max}	$\lambda_{\max} - \lambda_{\text{mid}}$

umns emphasize a somewhat surprising correspondence between the cardinality of the feature and the eigenvalues.

As usual, a disclaimer is in order. The above heuristic approach is by no means the only (or the best) indicator of saliency. We believe that it is one of the simplest to implement, is fairly intuitive, and proves to behave well as an indicator of saliency.

5.2.4.2 Combination per site for the Point field - 3-D case

when only 3-D points are available (no orientation), we first attempt to find a maximum likelihood normal to those points, with the aid of the Point field. Again we compute the 3 eigenvalues and eigenvectors, but a different interpretation is now needed. Recall that we selected a vector lying along the two points to represent all possible surfaces.

Recovering surface normals requires merely to select the eigenvector corresponding to the smallest eigenvalue. This orientation will be the vector perpendicular to the best plane described by the ellipsoid of votes.

In order for a certain location to be a good candidate for a surface, the votes have to be distributed in such a way that they create a “flat” sphere. This can obviously be tested by looking at $\lambda_{\text{mid}} - \lambda_{\min}$. Large values of that term will indicate high likelihood of a surface passing through the location.

The above procedure assigns orientations to the given set of input points. It could also assign strength to points, thus reducing the influence of noise *before* the second pass.

Unfortunately, it is impossible at this stage to recover intersections and junctions from the computed maps. It is necessary to perform a second convolution using the Patch Extension field, on the input data, which now has orientation data available. Obviously, the final results when a cloud of non-oriented points is given, are not as good as with oriented input data.

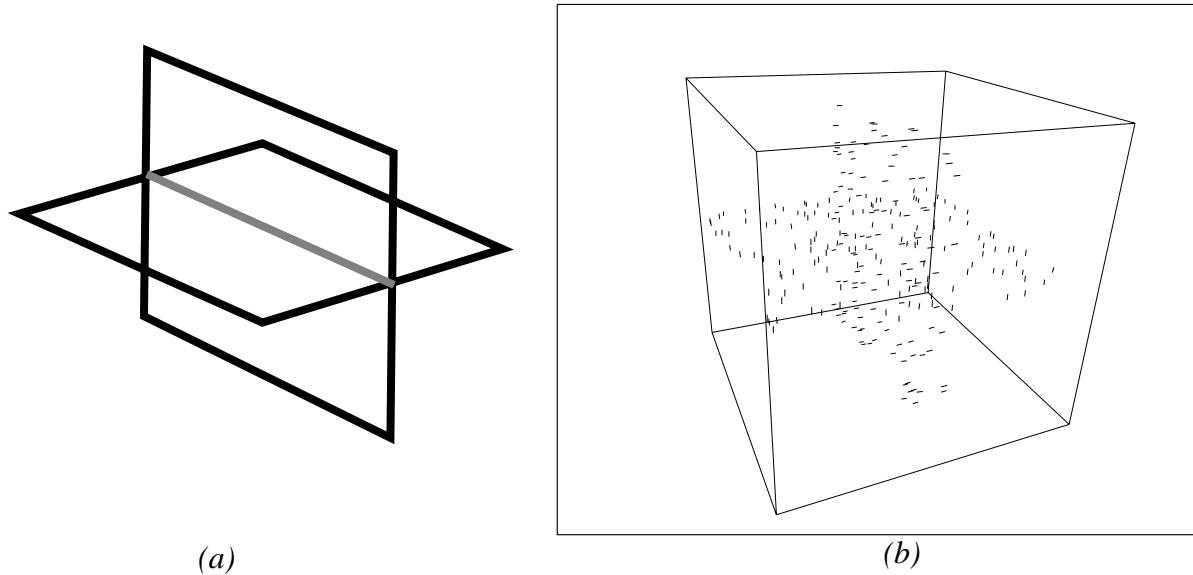


Figure 5.6 Input consists of randomly selected set of points lying on the two intersecting surfaces. (a) A schematic model of input (The lighter line denotes the intersection between surfaces). (b) Projection of input samples.

5.2.5 Noise tolerance

Similar to the 2-D case, the scheme is not sensitive to noise in the form of erroneous features, or localization errors of the measurements, since a voting scheme is employed. Also, a priori distribution of noise is expected to be directionally uniform, such that computed orientations are not corrupted. Saliency selectivity⁶, however, could suffer when noise is present.

5.3 Results

We have generated some synthetic images to test our scheme. The first example consists of two planes positioned in space as shown in Figure 5.6 . We randomly sampled the planes of Figure 5.6 . Grid size was 50X50X50, each plane has ~100 samples. Since 3D saliency maps are 4D in nature⁷, we thresholded (for display purposes) all maps to a point where the 2D projection becomes legible, and small line segments denote the orientation at each site In practice one would like to follow the dense saliency maps, and extract a description of each surface⁸. The task of following the surfaces along the saliency maps was not preformed in this work. Figure 5.7 shows the saliency maps for surfaces and curves. Note that the maps are dense now, and every site contains a normal. Also, the orientation of the segments in the intersection map (Figure 5.7 (b)) are pointing in the direction of curve.

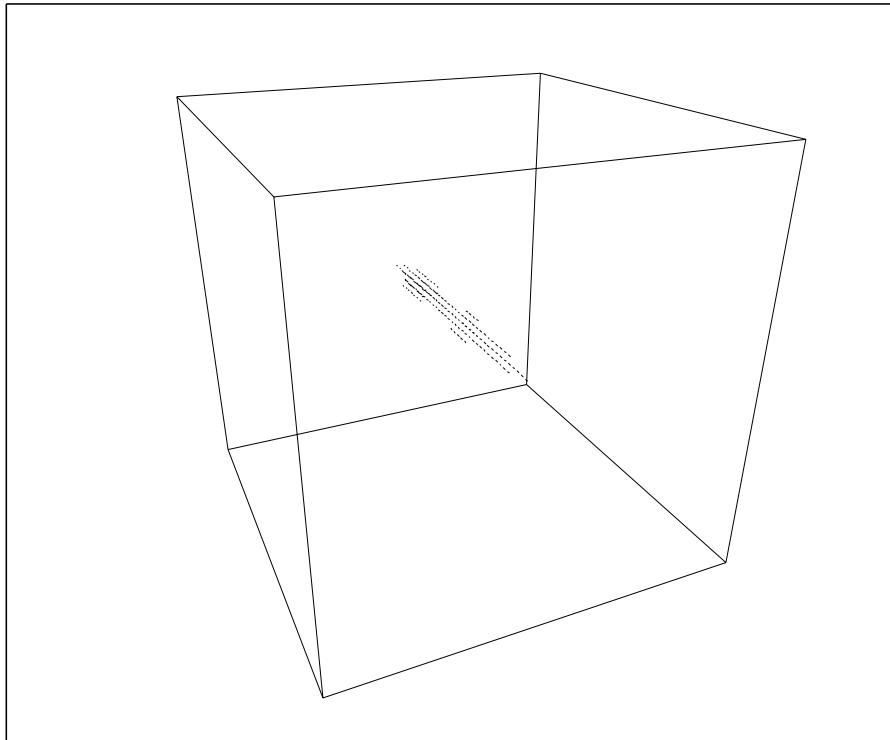
6. i.e., the variance between figure and ground in the saliency maps.

7. Strength and orientation at each 3D site.

8. e.g. by a triangulation.



(a)



(b)

Figure 5.7 (a) surface saliency map. (b) Curve saliency map.

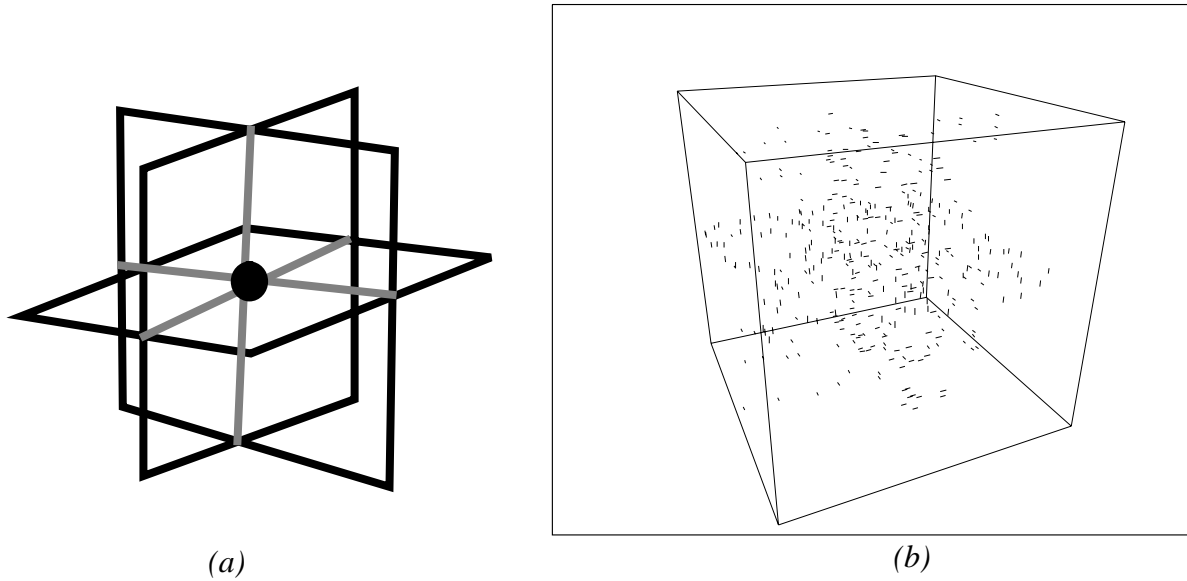


Figure 5.8 (a) A schematic model of input (The brighter lines denote the intersections between surfaces, and the dot is the 3D junction). (b) Projection of

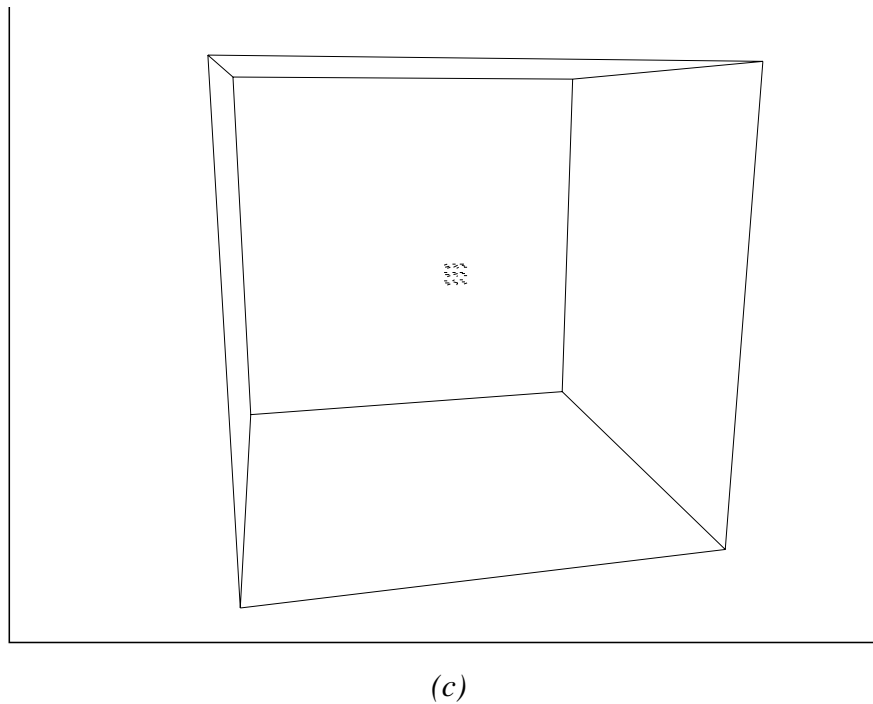
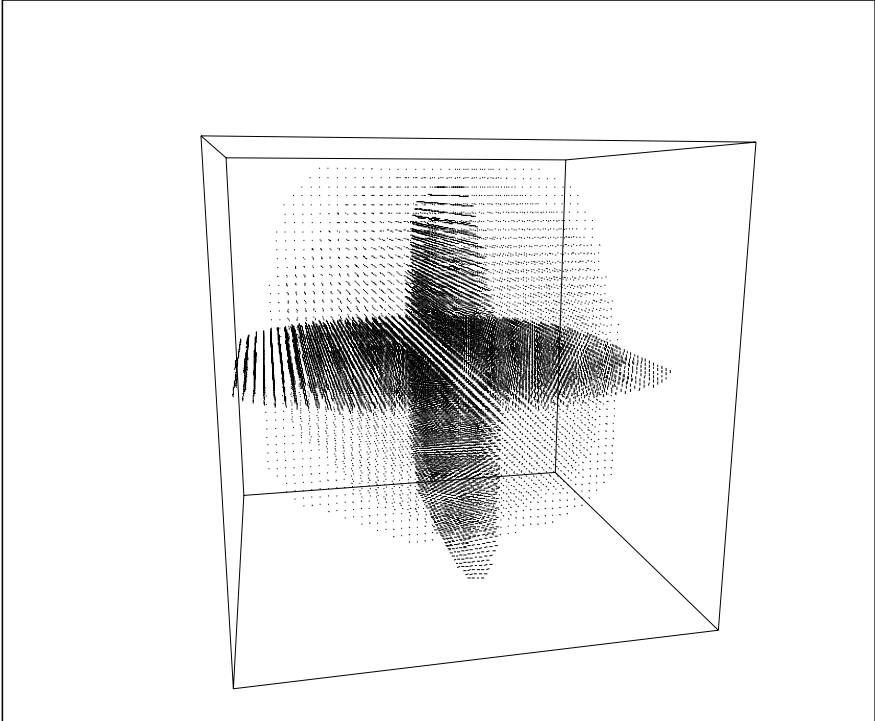
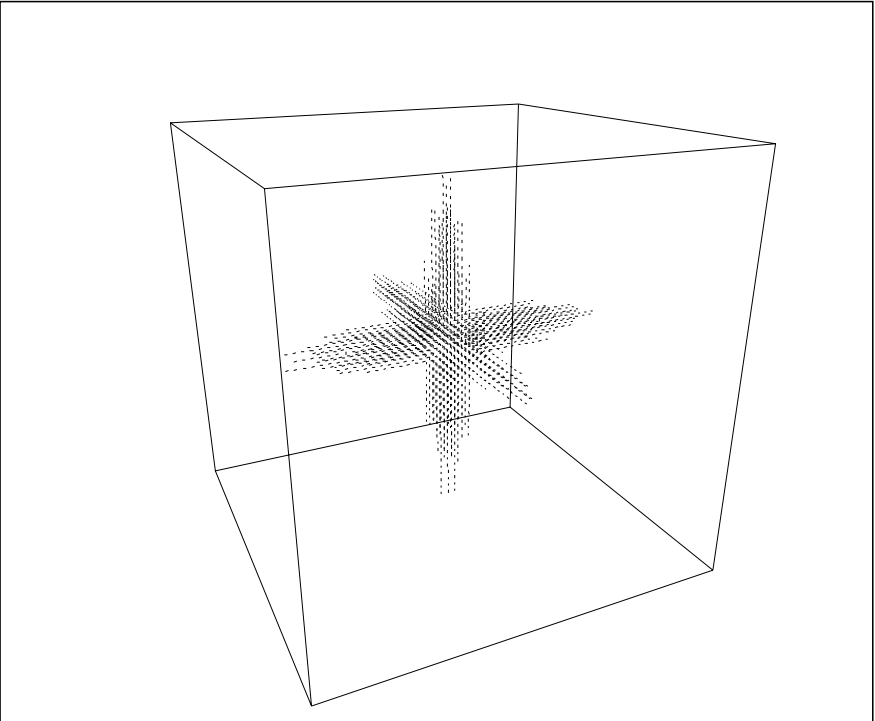


Figure 5.9 (continued)(c) Junction saliency map.

Figure 5.6 describes a scenario with three intersecting planes. Figure 5.9 shows the three corresponding saliency maps. Both the surface map and the intersection map tend to decay toward the edges of our 3D space. This is due to the limitations



(a)



(b)

Figure 5.9 (a) surface saliency map. (b) Intersection saliency map.

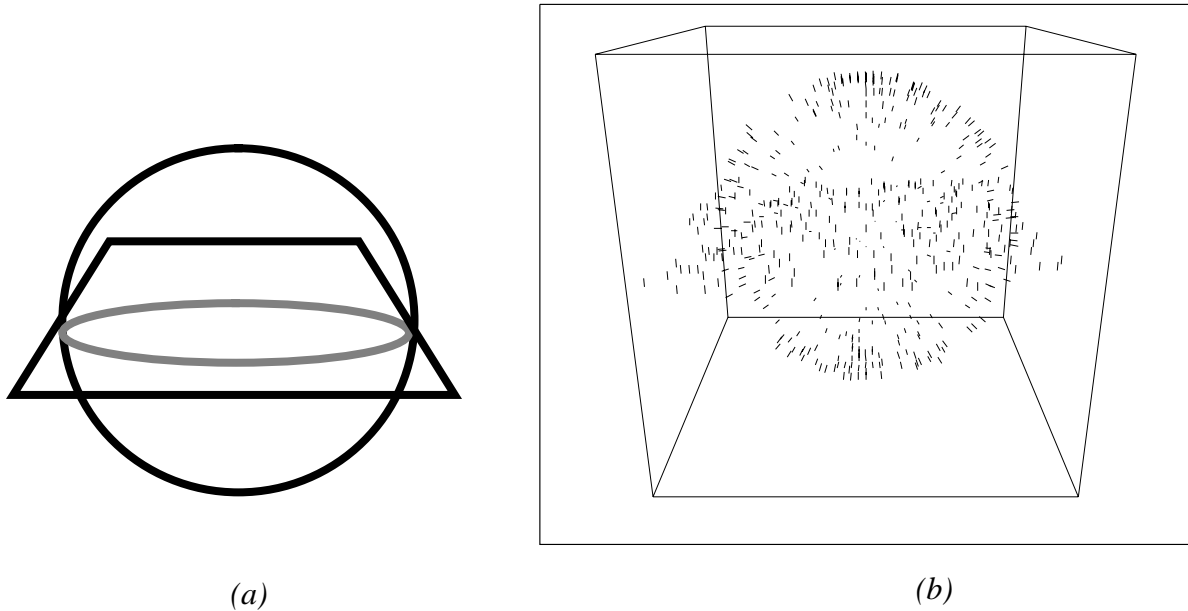


Figure 5.10 (a) A schematic model of input (The brighter lines denote the intersections between surfaces). (b) Projection of input samples.

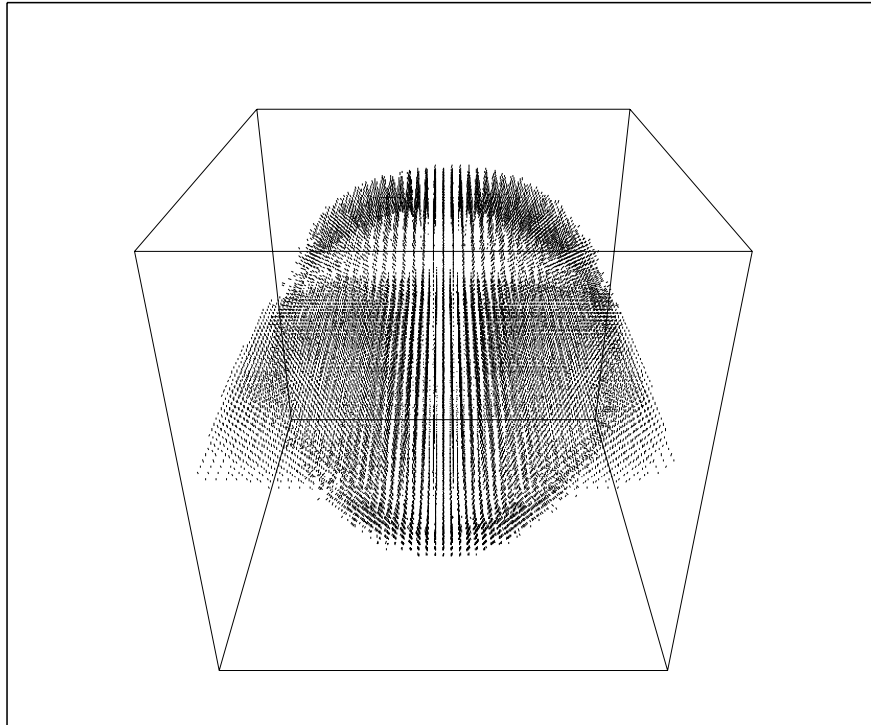
of our display and the application of a constant threshold to the data. Figure 5.10 depicts an example with curved surfaces. Here a sphere is intersected with a plane. As before, the plane consists of ~ 100 measurements, and the sphere has about 200 measurement points. Results are shown in Figure 5.9 .

5.3.1 Noise tolerance

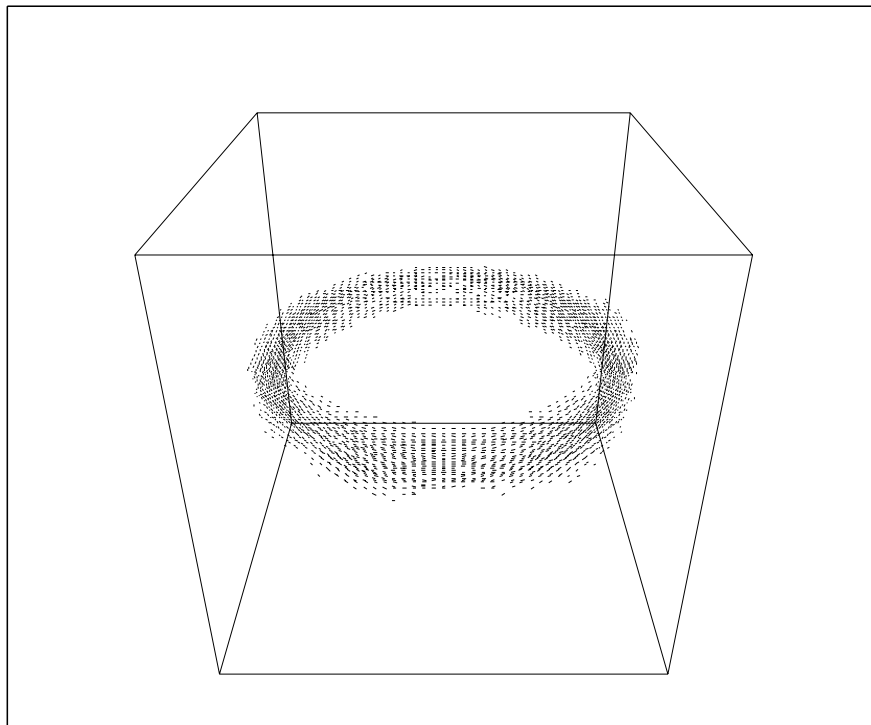
We choose a simple curved surface to illustrate the noise immunity of the scheme. A part of a sphere is chosen, and ~ 150 points are randomly selected on the sphere, as shown in Figure 5.12 . We ‘sprinkle’ the space with an increasing number of erroneous segments. The results in Figure 5.7 show the surface saliency maps with 125, 250, and 375 additional random segments. It is easy to see that they are virtually the same. The input set with 250 noisy points is shown in Figure 5.12 (b) for reference.

Finally, we show an example where the input consists of a cloud of non-oriented points, with a considerable amount of noise. Again a quarter of a sphere is embedded in noise. The sphere has ~ 200 data points, and ~ 100 noise points, as shown in Figure 5.14 (a).

The first phase is to compute normals to the existing input points. This is done by convolving with the 3D Point field. Figure 5.14 (b) shows the result of the first phase. Note that not only do the points have orientation vectors attached to them, but many of the noise points have been attenuated. The second phase is the standard Patch extension field convolution. The final result is shown in Figure 5.14 (c).

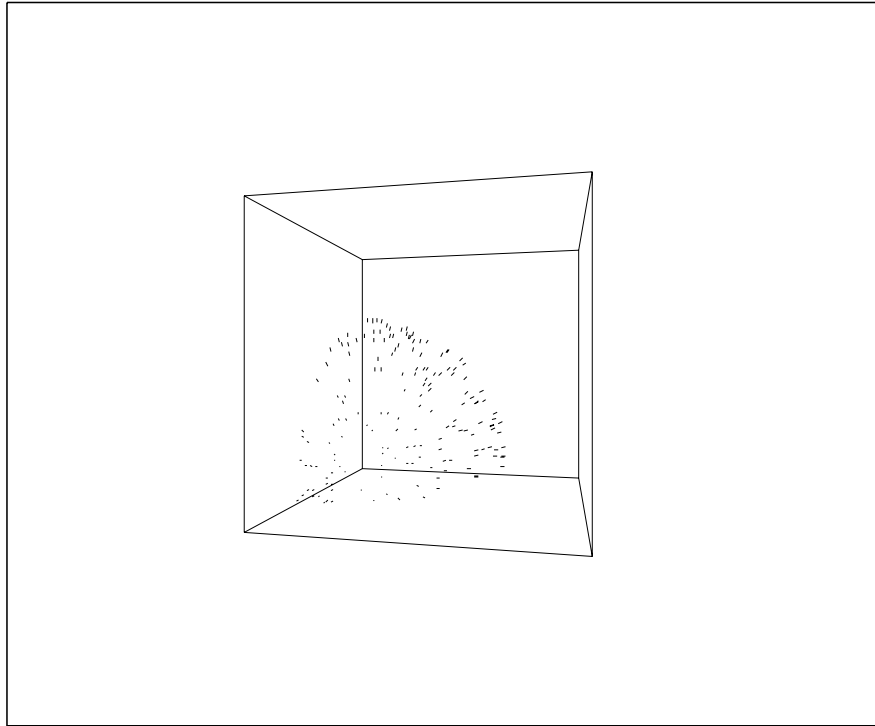


(a)

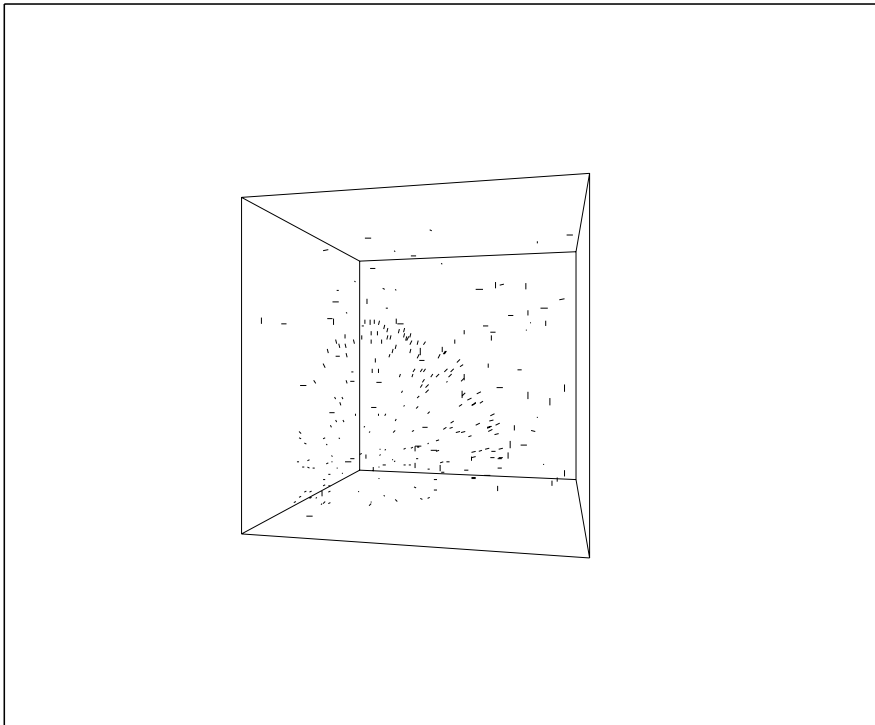


(b)

Figure 5.11 (a) surface saliency map. (b) Intersection saliency map.

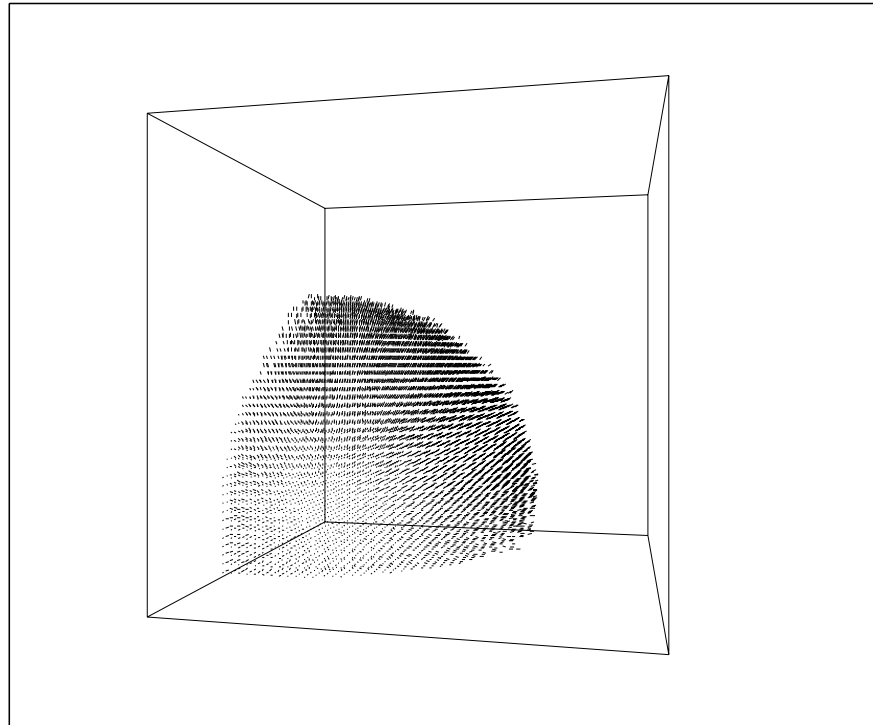


(a)

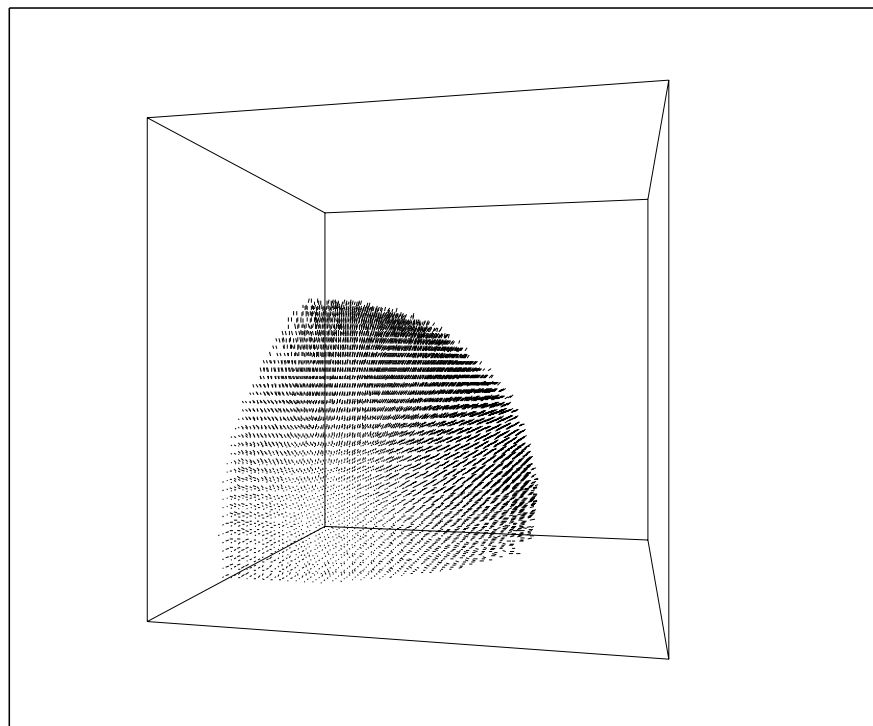


(b)

Figure 5.12 (a) Sample points of a quarter sphere centered at the origin.(b) Same sphere embedded in 250 noisy points.

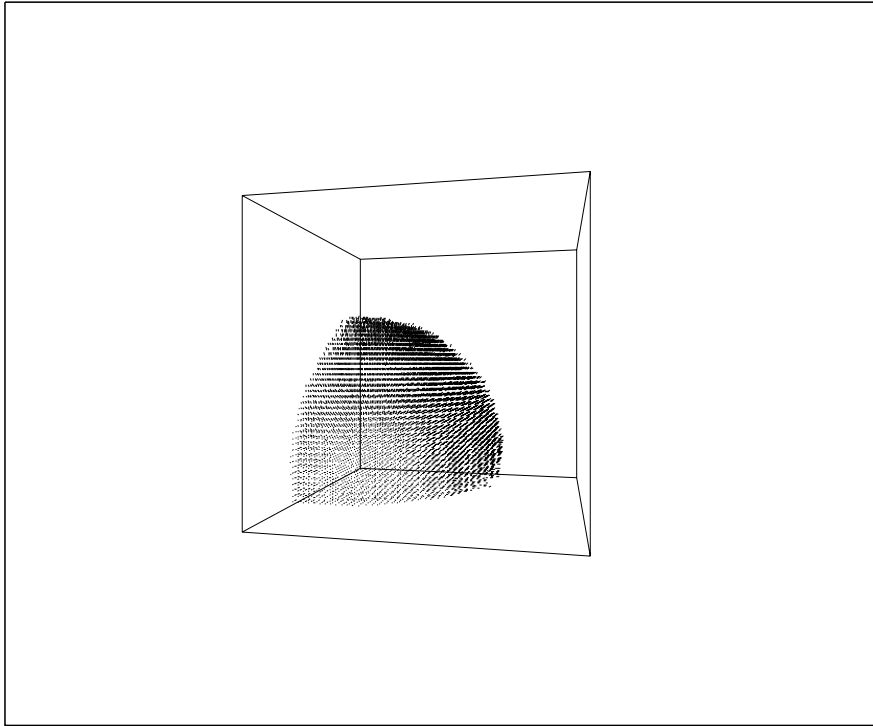


(a)

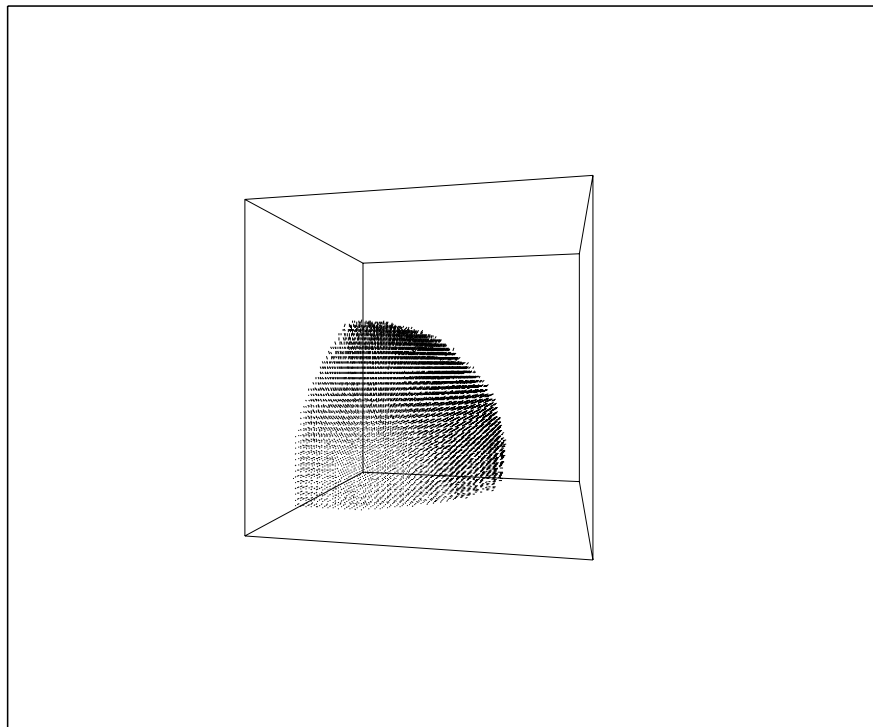


(b)

Figure 5.13 (a) surface saliency map without noise. (b) with 125 noise segments.

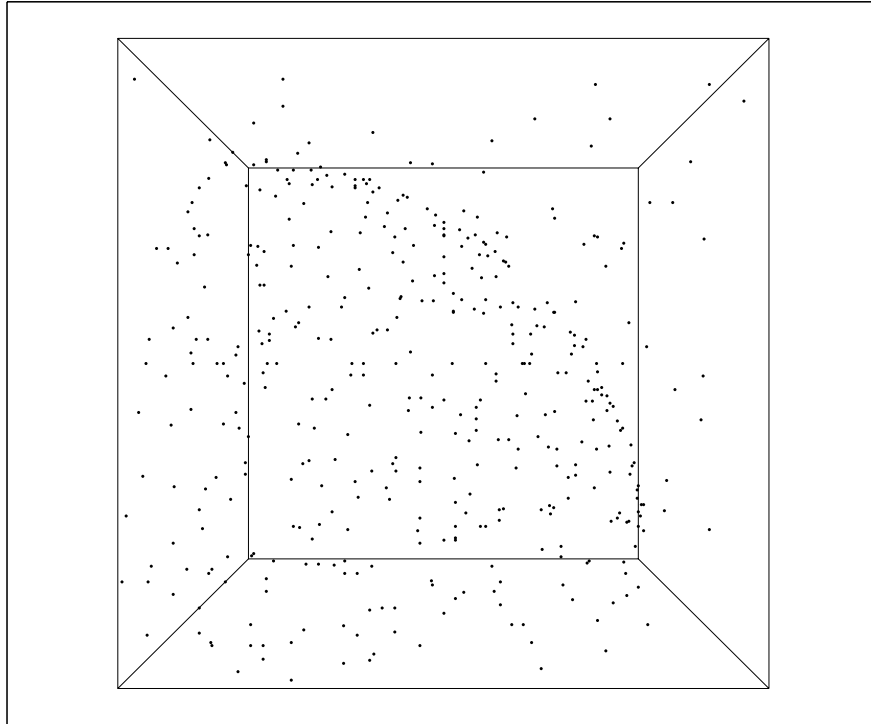


(c)

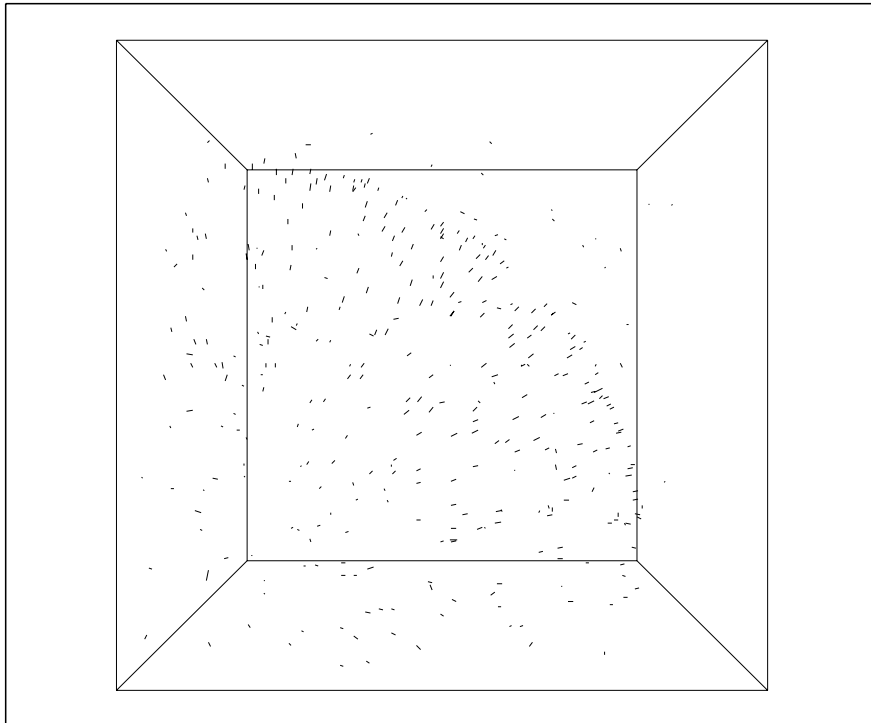


(d)

Figure 5.13 (continued) (c) with 250 segments. (d) with 375 noise segments.

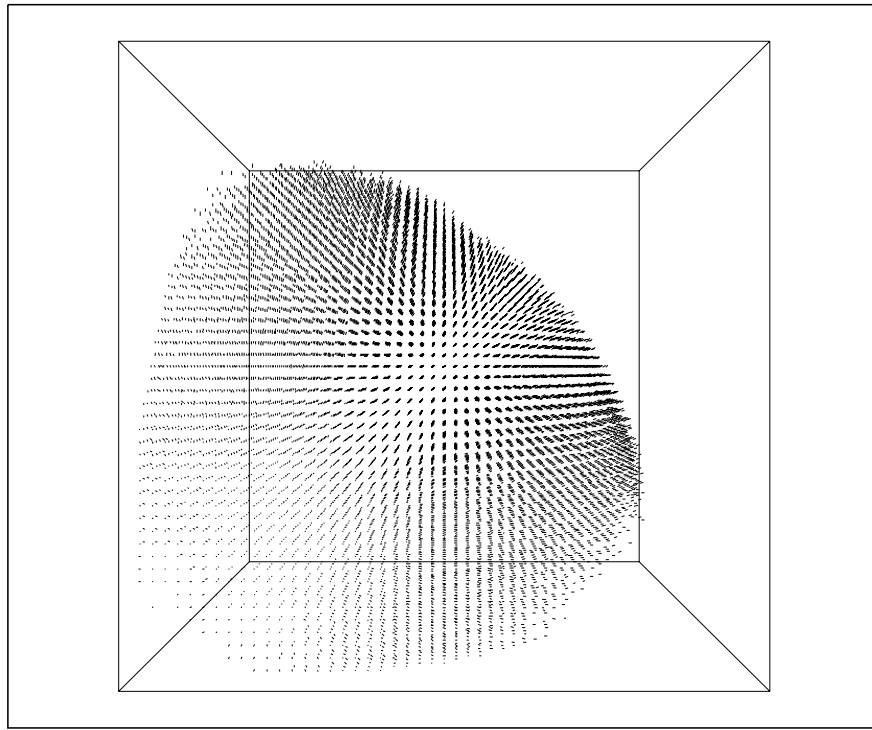


(a) Input Image



(b) With estimated normals and strength

Figure 5.14 Results of saliency analysis



(c)

Figure 5.14 (continued) (c) Final saliency map

5.4 Conclusion

We have presented a method to recover surfaces, intersection between surfaces, and 3-D junctions by applying perceptual grouping rules. The method presented is an extension of a 2-D approach proposed earlier by the authors, and uses a non-iterative and parameter-free algorithm. The method can handle scenes with any number of objects, each having an arbitrary genus number, without any a priori knowledge. In particular, an initial guess is not needed.

The complexity is $O(n^3k)$ in general, where n is the side size of the volume, and k is the number of available measurements. Some practical short-cuts can reduce the complexity further. The algorithm is highly parallel in nature, and as such can be easily implemented on a parallel machine.

5.5 References

- [59] N. Ahuja and M. Tuceryan, *Extraction of early perceptual structure in dot patterns: integrating region, boundary, and component Gestalt*, CVGIP 48, 1989, pp. 304-356.

- [60] J. Dolan and R. Weiss, *Perceptual Grouping of Curved Lines*, Proc. IUW89, Palo Alto, CA., pp. 1135-1145.
- [61] P. Fua and P. Sander, *Segmenting Unstructured 3D Points into surfaces*, ECCV 92, Santa Margherita Ligure, Italy, May 1992, pp. 676- 680.
- [62] G. Guy and G. Medioni, *Perceptual Grouping using Global Saliency enhancing operators*, Proc. of ICPR92, The Hague, Holland, 1992, pp. 99-104
- [63] G. Guy and G. Medioni, *Perceptual Grouping using Global Saliency enhancing operators*, IRIS-USC Technical report, *to appear*
- [64] G. Guy and G. Medioni, *Inferring Global Perceptual contours from Local Features*, Proc. of CVPR93, New-York, New-York, 1993, pp. 786-787.
- [65] M. Kass, A. Witkin, and D. Terzopoulos, *Snakes: Active Contour Models*, in *International Journal of Computer Vision*, January 1988, pp.321-331.
- [66] C. Liao and G. Medioni, *Surface Approximation of a Cloud of 3-D Points*, CAD94 workshop, Pittsburgh, PA.
- [67] D.G. Lowe, *Three-dimensional object recognition from single two-dimensional images*, *Artificial Intelligence* 31, 1987, 355-395.
- [68] R. Mohan and R. Nevatia, *Segmentation and description based on perceptual organization*, Proc. CVPR, Jun. 1989, San Diego, Ca., pp. 333-341.
- [69] R. Mohan and R. Nevatia, *Using Perceptual Organization to Extract 3-D Structures*, *IEEE Trans. on PAMI*, Vol. 11, No. 11, November 1989, pp. 1121-1139.
- [70] D. Terzopoulos, A. Witkin, and M. Kass, *Constraints on deformable models: Recovering 3D Shape and Nonrigid Motion*, *Artificial Intelligence*, Vol. 36, 1988, pp. 91-123.