# A Framework for Exploring High-Dimensional Geometry 

Sidharth Thakur ${ }^{\star}$ and Andrew J. Hanson ${ }^{\star \star}$<br>Computer Science Department, Indiana University Bloomington


#### Abstract

To extract useful information from high-dimensional geometric or structural data, we must find low-dimensional projections that are informative and interesting to look at. The conventional, manual-interaction methods used for this purpose are ineffective when the dimensionality of the data is high, or when the geometric models are complex. Standard methods for determining useful low-dimensional views are either limited to discrete data, or to geometric information embedded in at most three dimensions. Since geometric data embedded in dimensions above three have distinct characteristics and visualization requirements, finding directly applicable techniques is a challenge. We present a comprehensive framework for exploring high-dimensional geometric data motivated by projection pursuit techniques. Our approach augments manual exploration by generating sets of salient views that optimize a customizable family of geometry-sensitive measures. These views serve to reveal novel facets of complex manifolds and equations.


## 1 Introduction

Humans are adept at recognizing structural relations and patterns among data points, and the shape and geometric features of objects embedded in up to three spatial dimensions. Visualization methods attempt to exploit this perceptual capability to facilitate understanding and communicating complex information. Some common examples are visualizations of vector and scalar fields, state space diagrams of dynamical systems, high-dimensional geometry, and information visualizations that map multi-variate information onto geometric structures such as height maps, graphs of networks, and geometric primitives.

However, when the dimensionality of the information space increases beyond three, it is difficult to meaningfully represent all the relevant attributes of data such as structural patterns or geometric relations. The problems in visualizing and interpreting highdimensional information arise because the nature of high dimensions is unintuitive and reducing the dimensions of the data for inspection in the familiar dimensions two and three inevitably results in loss of information. This problem is particularly relevant to geometric information because the distortions of structural relations, such as those due to occlusions of important features, or apparent but false intersections of lines and surfaces in the projection, can make it difficult to recover the properties of the models being inspected.

[^0]Facilitating the ability to identify key aspects of high-dimensional geometric data ${ }^{1}$ is indispensable in many applications and domains of science. For example, while exploring projective and Riemannian geometry, an important consideration is the ability to recognize the global and local object features of complex geometric objects. This is crucial, for example, to understand salient topological properties. Other applications that involve user-intensive activities, such as goal-directed exploration, require tailored approaches for navigating and exploring the associated high-dimensional information spaces.

Although the visualization interfaces for multi-variate data and geometric information in two and three dimensions have matured considerably, high-dimensional geometryspecific issues and requirements have received only limited attention. As a result, the tools for exploring high-dimensional geometry still employ either two-variable combination displays or manual selection methods, which can be tedious and ineffective for exploring complex objects in very high dimensions. In this work, we attack the problem of finding efficient and perceptually-salient goal-directed methods suitable for exploring high-dimensional geometric models; we are motivated by the tremendous potential payoff implicit in exploiting high-performance graphics to make the world of highdimensional geometry accessible to experts and novices alike.

## 2 Background

The exploration and recovery of structure in high dimensions has been pursued actively to investigate multi-variate data, multi-dimensional functions and high-dimensional geometry. Investigations in all these domains are concerned with the preservation and discovery of structural relations in the dimensionally reduced representations.

Multi-variate data analysis is usually concerned with the exploration of the most basic data primitive, the point, which represents a vector of information variables. Popular methods used for exposing clusters and groupings of the data in lower dimensions are PCA [1], Euclidean distance-preserving methods like MDS [2], methods based on random projections [3,4], and tour-based methods that employ a series of space-filling projections [5-7]. A particularly important and useful approach is called projection pursuit [8,9], which involves the computation of the information content in some lowdimensional projection using a numerical measure (or projection index).

The domain of multi-dimensional functions focuses on visualizing higher-order primitives, such as manifolds and hypersurfaces occurring in the visualization of physical simulations, mathematical constructions, and complex dynamical systems [10-15].

High-dimensional geometry, on the other hand, is concerned with the exploration of mathematical models embedded in high spatial dimensions. The fascination with geometry in high dimensions can be traced back to Abbott's classic treatment in Flatland [16]; however, modern efforts to graphically represent high-dimensional models are usually attributed to Noll's proposed methods for hyper-objects [17], and Banchoff's visualizations of classical two-manifolds embedded in four dimensions [18, 19].

[^1]Since then, extensive research has focused on developing tools and interfaces for exploring geometry and structure in three [20,21], four [22-26] and higher-dimensional spaces [27]. Some important contributions of these investigations are 4D lighting models [28-30], interfaces for manipulating high-dimensional objects [31,32], and the recent physically-based and multi-modal explorations of the fourth dimension [33-35].

Our contribution to visualizing high-dimensional geometry diverges from previous work in that we combine major concepts and methods of projection search and exploration with perception-motivated user interaction tools. The foundation of our approach is the fact that geometric data and their projections have inherent structural relations that can be exploited to discover views that are useful both for research-motivated exploratory analyses and for pedagogical, explanatory demonstrations.

## 3 Framework for Exploring Geometry

Our framework for exploring geometry in high dimensions is based on a variant of projection pursuit that we will refer to as geometry projection pursuit (GPP). The goal of GPP is to find projections of geometric models that reveal their important characteristic features ${ }^{2}$. Our framework for GPP includes two important aspects:

- Optimal view finding. We are concerned with finding useful low-dimensional views of a given geometric model by optimizing appropriate geometric projection indices. A combination of one or more indices can be used based on the goals of the exploration task or the nature of the geometric information. Section 4 discusses optimal view finding and index construction in detail.
- Visualizing the search space. Our framework includes the exploitation of displays that might be called "meta-projections" of the measures plotted against the projection variables themselves. This rich visualization is composed of the displays of the measure of the projection indices in all the possible sub-space projections. These meta-landscapes are employed to represent the richness of various subspaces, and are treated in Section 5.

Figure 1 summarizes the important steps involved in the process. The framework integrates manual exploration in $N$-dimensions with optimal view selection. Selecting optimal views involves either optimizing a set of projection indices or exploring a visualization of the projections in all relevant subspaces.

### 3.1 Geometry-specific issues addressed by the Framework

The distinct nature of geometric information compared to other kinds of data in high dimensions dictates specific requirements for visualization and GPP. Our framework addresses two important issues concerning high-dimensional geometry:

[^2]

Fig. 1. Framework for visualizing high-dimensional geometry: after loading a geometric model into th visualization interface. The geometry can be explored manually or by index optimization with GPP. GPP assists exploration by automatic generation of optimal views or via meta visualizations, which are snapshots of index measures in plotted in parameter subspaces.

- Nature of Geometric Information. Unlike point sets of multi-variate data, geometric data are composed of vertices that are associated in some order to form simplices, geometric primitives that describe an elementary $p$-dimensional region using $p+1$ vertices; geometric data have structural information inherent to the constituent data points. Geometric models are typically represented by geometric primitives (simplices) of increasing order of complexity, such as points (vertices), lines (edges), and surfaces (polygons). An important implication of this organization is that the standard approaches to multi-variate data analysis, such as clustering or 'dense' views, are no longer generally applicable due to overlap or occlusion in the projections of the higher order primitives. A typical example is the toroidal 2-manifold embedded in 4D shown in Figure 2: the torus is represented using primitives of increasing complexity but decreasing ambiguity ${ }^{3}$.


Fig. 2. A torus embedded in 4D, displayed using different orders of simplices as geometric primitives. The vertices of the torus are obtained by the parametric equation: $\mathbf{x}=$ $(\cos (\theta), \sin (\theta), \cos (\phi), \sin (\phi))$, where $\theta, \phi \in[0,2 \pi]$. The different primitives are: (a) points, (b) edges, (c) surfaces, and (d) all together.

- Specific tasks in GPP: The design of projection indices used in projection search must be sensitive to some specific requirements in GPP. For example, an important task in GPP is to identify salient topological features of an object. An example of a

[^3]particularly interesting feature is the number of holes in a 2-manifold, which indicates the object's genus. Figure 3 shows a sphere (with zero holes), and orthogonal projections of 3D and 4D embeddings of the torus, each with a single hole. Other relevant features include symmetry, occlusion and branch-cuts.


Fig. 3. Holes in geometric objects: (a) A sphere with no holes (genus=0). (b) A 3D-embedded torus with one hole (genus=1). (c) A 4D-embedded torus also with one hole (genus=1). (d) Shows the 4 D -embedded torus in a projection for which the hole is no longer visible.

## 4 Optimal View Finding in N-Dimensions

In this section we describe the first important component of our framework for GPP: enumerating appropriate information measures and finding one or more low-dimensional views that have the maximum amount of information content.

Traditionally, the problem of defining a "good view" of some 3D object or a complex scene is encountered frequently in object representation and recognition research. Traditional methods employ both image-based and non-image based rendering techniques to determine scene complexity and the overall information in the projected views or silhouettes of objects [36,37]. Other approaches are derived from object and shape perception literature, which include contrasting models for either 2D or 3D shape encoding (see [37]).

In our scope of work, good or desirable views correspond to the projections (usually in 2D or 3D) that reveal some desired property of the objects embedded in $N$ spatial dimensions. Unlike viewpoint selection in three dimensions, our problem is two-fold, in that not only does a good viewpoint need to be found in $N$ dimensions, but a guideline for suitable projection is required that maximizes an information measure (projection index) in some target projection dimension. In our case the choice of the projection index is usually dictated by the nature of the geometric information or the goal of the exploratory task.

### 4.1 Geometric Indices

A projection index is a function that computes a numerical value corresponding to the total information content in some projection. Projection pursuit in multi-variate data analysis typically makes use of indices that are based on statistical or entropy-based calculations. Appropriate indices for geometry exploration, however, can (and should)
exploit geometry-specific information inherent in the data. Below, we provide a list of representative geometry-based indices, which is by no means exhaustive and can be extended to include other meaningful representations:

Table 1: Examples of projection indices suitable for GPP

| Geometric Primitive | Indices |
| :---: | :---: |
| Points $=0$-simplices | The most straightforward set of indices are point spread and point-point distances. Methods used for discrete data points are also applicable. Given a set of $N$-dimensional points, $P=$ $\left(p_{1}, p_{2}, . ., p_{m}\right)$ we define: $\begin{aligned} I_{\text {spread }} & =\sum\left(\left\|p_{i}\right\|\right) \\ I_{\text {point_distances }} & =\sum_{i \neq j}\left(\left\|p_{i}-p_{j}\right\|\right) \end{aligned}$ |
| Edges $=1$-simplices | Edges provide our first measure that directly represents topological structure of a geometric object. Some suitable measures are: $\begin{aligned} I_{\text {edge_intersection }} & =\sum F\left(e_{i j}, e_{m n}\right) \\ I_{\text {edge_length }} & =\sum_{i \neq j} e_{i j} \end{aligned}$ <br> where $e_{i j}$ and $e_{m n}$ are edges between pairs of distinct vertices $i, j$ and $m, n$; the function $F\left(e_{i j}, e_{m n}\right)$ determines if there is intersection between the two edges. |
| Faces $=2$-simplices | 2-simplices represent a two-dimensional manifold or surface patch in terms of triangles. A simple measure given the surface elements represented by the vertex triplets $(i, j, k)$ is: $I_{\text {surface_area }}=\sum \text { Area }_{i j k}$ <br> Some other suitable measures are those which compute the number of occlusions based on 3D depth buffer hits, or face/face intersections in the projections. |
| Volume (or Balls) $=$ 3-simplices | A 3-simplex is a three-dimensional volume element. Projection criteria can be selected to minimize/maximize the projected volume, or compute the occlusions based on 3D or 4D depth buffer hits. |
| $\begin{aligned} & \text { Hypervolume = 4- } \\ & \text { simplices } \end{aligned}$ | Interesting views can correspond to projections that maximize the polygonal surface area or volume in the 4 D projection, which can be explored in special tools that allow full 4D rotations. This is distinct from maximizing the 2D or 3D projections of these quantities. |

Table 1: Examples of projection indices suitable for GPP

| Geometric Primitive | Indices |
| :--- | :--- |
| Non-geometric <br> indices | Image-based rendering methods can be combined with the <br> above-mentioned geometric indices to obtain some suitable <br> metric. For example, we have a hole-finding technique that <br> utilizes pixel-color information to identify holes of arbitrary <br> shapes and sizes. |
| Space <br> Method | spalking <br> mace walking [38] is a method for navigating on topological <br> onto the screen plane. This solution is similar to a local pro- <br> jection pursuit approach because, at each step, as the observer <br> rotates the model, the control system determines the transfor- <br> mation for orienting the local tangent space around the point of <br> interest parallel to the screen display plane. |

Once one or more indices are chosen, we use the procedure for GPP shown in Figure 4. The step involving viewpoint selection in the $N$-dimensional space can be achieved by different methods. For example, random configurations of the $N \times N$ rotation matrix corresponding to random rotations can be evaluated for optimal views. Another method is to utilize an optimizer to find the best combination of high-dimensional rotations.


Fig. 4. Finding optimal views by GPP. First, a candidate target projection space is specified. Next, different projections in the target space are obtained by repeatedly performing rotations in $N$ dimensions and computing the measure of the indices. Finally, the projections with high values of the indices are selected.

## 4.2 $N$-Dimensional Optimization methods

Once a set of indices has been specified, optimizations in the high-dimensional space are performed to find a series of maxima (or minima, depending on the type of index). The general optimization problem can be formalized as follows:

ND optimization problem: Given a function $f\left(\theta_{i}, i=1,2, \ldots, N(N-1) / 2\right)$ that accepts $N(N-1) / 2$ rotation angles, one in each of the canonical two-dimensional planar subspaces spanned by two orthogonal axes, and that returns the value of a projection index, find the set of orientation angles $\left\{\theta_{i}^{\prime}\right\}$ such that for all $\theta_{i} \in$ $[0,2 \pi]$, the function satisfies $f\left(\left\{\theta_{i}^{\prime}\right\}\right) \geq f\left(\left\{\theta_{i}\right\}\right)$.

An important consideration in the ND optimization problem is that search in an angle $\theta_{i}$ must be restricted to transformations that actually effect a change in the chosen projection space. Furthermore, as the dimensionality of the embedding space increases (e.g., $N \geq 5$ ), the full optimization tends to run very slowly, and finding all local optima in the search space is a challenge. The problem can be approached, e.g., by simulated annealing but that can also be very time consuming. While we plan to implement largesubspace optimization in future approaches, here we focus on an effective approximate sequential search method using low-dimensional subspaces. Our procedures for $N(N-$ 1) $/ 2$ one-dimensional subspace search and pairwise searches of the $N(N-1)\left(N^{2}-N-\right.$ 2)/8 two-dimensional subspaces of the $\theta_{i}$ are thus as follows:

1D (2D) approximate optimization method: Compute the optimization measure $f\left(\left\{\theta_{i}\right\}\right)$ for one choice of $i$ (or $(i, j)$ ), and find the optimal point. Fix that point and repeat for the next $i($ or $(i, j)$ ). Repeat. The result will deviate from a true optimum but will give good estimate for beginning a less time-consuming search in the neighborhood of the resulting point. Variants include performing local subspace optimization incrementally in neighboring subspaces.
utilize the Conjugate Gradient method for finding the local optima. We have found that almost-optimal views, or local optima, are equally interesting as they may reveal additional perceptual aspects of a geometric model that the mathematically optimal views may obscure. We therefore track all local minima found by the algorithm. While we have not accumulated extensive performance data, computing the 2D subspaces of a 2500-vertex surface in 9D takes about 3 minutes; further performance optimization strategies are being studied.

### 4.3 Results of Optimal View Finding

We now discuss some results obtained by applying our methods to objects in high dimensions. Figure 5 shows different views of a pillow-shaped 4D object obtained by maximizing the area of the object projected into three dimensions.

Figure 6 contrasts the optimal views of a 4D-embedded torus based on two different projection indices: (a) projection edge length, and (b) projection area.

Figure 7 shows a more complex example, the cubic $C P^{2}$ torus, which consists of nine two-dimensional parametric patches embedded in nine dimensions. Figure 7(b) shows an interesting projection in the 3D target space with axes $(5,7,9)$. The optimal view finding method assisted in discovering the object's single hole (genus=1) ${ }^{4}$.

[^4]

Fig. 5. Different projections of a 4D pillow-shaped object whose parametric equation is $\{\cos (\theta), \cos (\phi), \sin (\theta) \sin (\phi), \sin (\theta) \sin (\phi)\}$. The projections into the 3D space with axes $(1,2,3)$ are: (a) The default view (b) A flattened view. (c) An optimal view based on a maximal area projection found by rotations in subspaces $(2,3)$ and $(2,4)$.


Fig. 6. Different projections of a 4D-embedded torus in the projection space with indices $(1,2,3)$ (a) The default view. (b) View found by maximizing edge-length between projected vertices; this view was found by rotations in subspaces $(1,2)$ and $(1,4)$. (c) A view based on maximal-area projection found by rotations in subspaces $(1,3)$ and $(1,4)$.

## 5 Visualizing the Search Space of Optimal Views

The second significant part of the framework is the visualization of the search space of all possible projections. We refer to this as the meta visualization because it provides additional information on the measures of projection indices. Figure 8 shows a snap shot of the visualization interface and the meta visualizations.

The meta visualizations provide a quick overview of the richness of different subspaces in the form of thumbnail charts as shown in Figure 8(b). The thumbnails are generated by automatically roaming through all the subspaces, e.g., by performing rotations in the subspaces and recording values of indices. For objects in very high dimensions, there are possibly hundreds of such thumbnails, and to speed up computation, the user can either reduce the resolution of the charts or specify what subspaces are to be explored.

The thumbnails can be magnified into higher resolution charts by clicking on them in the thumbnails panel. An example is shown in 8(c), in which the horizontal and vertical axes correspond to rotations in two different subspaces, respectively. Clicking and dragging anywhere on the magnified chart applies the corresponding rotations to the object in the main window. This allows quick investigation of optimal views and their neighborhoods. The chart in Figure 8(c) shows a well-defined region of maxima of the index; other meta visualizations can of course be much more complex. Another example in shown in figure 9 , which shows a 4 D torus in a projection that exposes its hole, and the meta visualization chart that depicts the associated subspace.


Fig. 7. Different projections of a two-dimensional manifold, actually a torus, embedded in ninedimensional space: (a) The default view in projection space with indices (1,2,3). (b) Default view in the projection with indices $(5,7,9)$. (c) A view in the space $(5,7,9)$ that exposes a hole; the view was found by maximizing the index based on distances between vertices in 3D, a point spread measure, and the area of the projected 3D surface. The exposure of the hole was maximized by exploring the neighborhoods of subspaces $(4,9),(6,9)$ and $(8,9)$.


Fig. 8. The visualization interface. (a) The visualization application. (b) The closeup of the thumbnail charts representing the measure of a projection index in all 2D subspaces. (c) A magnified view of the thumbnail for subspace $(1,3)$ and $(1,4)$ (thumbnail with black background). The palette corresponds to the values of the index: dark red $=$ high value, dark blue $=$ low value.

## 6 Conclusion and Future Work

We have presented a framework for visualizing and exploring high-dimensional geometry using a general philosophy that we refer to as geometric projection pursuit (GPP). We present several relevant examples of the method targeting the fact that understanding geometric objects in high dimensions has unique and specific requirements in terms of visualization and exploration. Projection search based on optimization of geometrydriven indices is useful in exposing features of the object that are relevant to their geometric and topological properties, and to goal-specific tasks requiring the elucidation of such properties. While we have given examples of some families of simple geometrybased projection indices, additional arbitrarily complex problem-specific measures can easily be added to our system. We plan to experiment further with indices that expose characteristics like symmetry, occlusion, branch-cuts and non-rigid deformations. It may also be possible to extend the framework to non-geometric data sets and thus to enhance techniques such as $N$-dimensional brushing [39]. Another extremely promising area for future work is the development of intelligent navigational methods to create smooth transitions among the extensive set of optimal views found by the system.

This research was supported in part by National Science Foundation grants CCR0204112 and IIS-0430730.


Fig. 9. (a) The hole in a 4D-embedded torus. (b) Its meta visualization in one 2D subspace, ( 1,4 ) combined with $(3,4)$. The dark red areas correspond to the hole in the screen plane. The hole was identified by comparing the pixel colors for the rendered background and foreground.

## References

1. Jolliffe, I.T.: Principal Component Analysis. Second edn. Springer Series in Statistics. Springer, New York, NY, USA (2002)
2. Kruskal, J.B., Wish, M.: Multidimensional Scaling. Sage University Paper series on Quantitative Application in the Social Sciences. Sage Publications, Beverly Hills and London (1978)
3. Dasgupta, S.: Experiments with random projection. In: UAI '00: Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence, San Francisco, CA, USA, Morgan Kaufmann Publishers Inc. (2000) 143-151
4. Fern, X., Brodley, C.: Random projection for high dimensional data clustering: A cluster ensemble approach. In: Twentieth International Conference on Machine Learning (ICML2003). (2003)
5. Asimov, D.: The grand tour: a tool for viewing multidimensional data. SIAM J. Sci. Stat. Comput. 6 (1985) 128-143
6. Cook, D., Buja, A., Cabrera, J., Hurley, H.: Grand tour and projection pursuit. J. of Computational and Graphical Statistics 2 (1995) 225-250
7. Young, F., Rheingans, P.: Visualizing structure in high-dimensional multivariate data. The IBM Journal of Research and Development 35 (1991)
8. Huber, P.J.: Projection pursuit. The Annals of Statistics 13 (1985) 435-475
9. Friedman, J.H.: Exploratory projection pursuit. J. of the American Statistical Association 82 (1987) 249-266
10. Mihalisin, T., Timlin, J., Schwegler, J.: Visualizing multivariate functions, data, and distributions. IEEE Comput. Graph. Appl. 11 (1991) 28-35
11. van Wijk, J.J., van Liere, R.: Hyperslice: visualization of scalar functions of many variables. In: VIS '93: Proceedings of the 4th conference on Visualization '93. (1993) 119-125
12. Liere, R., Wijk, J.J.: Visualization of multi-dimensional scalar functions using hyperslice. Technical report, Computer Science Department, CWI, Amsterdam, The Netherlands, The Netherlands (1994)
13. dos Santos, S.R., Brodlie, K.W.: Visualizing and investigating multidimensional functions. In: VISSYM 202, Aire-la-Ville, Switzerland, Switzerland, Eurographics Association (2002) 173
14. Simionescu, P.A., Beale, D.: Visualization of hypersurfaces and multivariable (objective) functions by partial global optimization. The Visual Computer 20 (2004) 665-681
15. Wegenkittl, R., Löffelmann, H., Gröller, E.: Visualizing the behavior of higher dimensional dynamical systems. In: VIS 1997, Los Alamitos, CA, USA, IEEE Computer Society Press (1997) 119-126.
16. Abbott, E.A.: Flatland: A Romance of Many Dimensions. Dover Publications Ltd. (1992)
17. Noll, A.M.: A computer technique for displaying n-dimensional hyperobjects. Commun. ACM 10 (1967) 469-473
18. Banchoff, T.: Discovering the Fourth Dimension. Prime Computer, Inc., Natick, MA (1987)
19. Banchoff, T.: Beyond the Third Dimension: Geometry, Computer Graphics, and Higher Dimensions. W. H. Freeman \& Co., New York, NY, USA (1990)
20. Phillips, M., Levy, S., Munzner, T.: Geomview: An interactive geometry viewer. Notices of the Amer. Math. Society 40 (1993) 985-988
21. Hanson, A.J., Munzner, T., Francis, G.: Interactive methods for visualizable geometry. Computer 27 (1994) 73-83
22. Hoffmann, C.M., Zhou, J.: Some techniques for visualizing surfaces in four-dimensional space. Comput. Aided Des. 23 (1991) 83-91
23. Hollasch, S.: Four-space visualization of 4D objects. Master's thesis, Arizona State University (1991)
24. Banks, D.: Interactive manipulation and display of surfaces in four dimensions. In: SI3D '92: Proceedings of the 1992 symposium on Interactive 3D graphics, New York, NY, USA, ACM Press (1992) 197-207
25. Roseman, D.: Twisting and turning in four dimensions (1993) Video animation, Department of Mathematics, University of Iowa, and the Geometry Center.
26. Hanson, A.J., Ishkov, K.I., Ma, J.H.: Meshview: Visualizing the fourth dimension. Technical report, Indiana University (1999)
27. Hanson, A.J. Graphics Gems IV. In: Geometry for n-dimensional graphics. Academic Press, Cambridge, MA (1994) 149-170
28. Hanson, A.J., Heng, P.A.: Visualizing the fourth dimension using geometry and light. In: VIS 1991, Los Alamitos, CA, USA, IEEE Computer Society Press (1991) 321-328
29. Carey, S.A., Burton, R.P., Campbell, D.M.: Shades of a higher dimension. Computer Graphics World (1987) 93-94
30. Steiner, K.V., Burton, R.P.: Hidden volumes: The 4th dimension. Computer Graphics World (1987) 71-74
31. Duffin, K.L., Barrett, W.A.: Spiders: a new user interface for rotation and visualization of n -dimensional point sets. In: VIS '94: Proceedings of the conference on Visualization '94, Los Alamitos, CA, USA, IEEE Computer Society Press (1994) 205-211
32. Hanson, A.J. Graphics Gems V. In: Rotations for n-dimensional graphics. Academic Press, Cambridge, MA (1995) 55-64
33. Hanson, A.J., Zhang, H.: Multimodal exploration of the fourth dimension. In: Proceedings of IEEE Visualization. (2005) 263-270
34. Zhang, H., Hanson, A.J.: Physically interacting with four dimensions. In: ISVC. Number 1 in 4291 (2006) 232-242
35. Zhang, H., Hanson, A.J.: Shadow driven editing (accepted for publication). In: Proceedings of IEEE Visualization. (2007)
36. Vázquez, P.P., Feixas, M., Sbert, M., Llobet, A.: Viewpoint entropy: a new tool for obtaining good views of molecules. In: VISSYM '02: Proceedings of the symposium on Data Visualisation 2002, Aire-la-Ville, Switzerland, Switzerland, Eurographics Association (2002) 183-188
37. Polonsky, O., Patane, G., Biasotti, S., Gotsman, C., Spagnuolo, M.: What's in an image: Towards the computation of the "best" view of an object. The Visual Computer 21 (2005) 8-10
38. Hanson, A.J., Ma, H.: Space walking. In: VIS '95: Proceedings of the 6th conference on Visualization '95, Washington, DC, USA, IEEE Computer Society (1995) 126
39. Martin, A.R., Ward, M.O.: High dimensional brushing for interactive exploration of multivariate data. In: VIS '95: Proceedings of the 6th conference on Visualization '95, Washington, DC, USA, IEEE Computer Society (1995) 271

[^0]:    * e-mail: sithakur@cs.indiana.edu
    ** e-mail: hanson@cs.indiana.edu

[^1]:    ${ }^{1}$ Terminology: geometric data and geometric models, are used interchangeably to refer to a geometric object; geometric information refers to the higher level construct that includes the geometric data and the embedding space.

[^2]:    ${ }^{2}$ Although the bulk of the literature on projection search and dimensionality reduction focuses on finding informative views in two dimensions, we are concerned with finding 'good' views in two, three, or even four dimensions, where the latter can be explored using efficient humanguided interactive techniques in 3D and 4D viewers.

[^3]:    ${ }^{3}$ The higher order primitives may not always disambiguate the geometric structure; for example, while projecting from from $N$ dimensions to two or three dimensions, false intersections may arise as artifacts of projection. Additional cues like depth-keyed coloring may be used to provide information on the relative depth of the intersecting segments.

[^4]:    ${ }^{4}$ While the properties of the object were known beforehand, optimal view-finding aided in finding the large hole, which is very tedious to discover manually.

