Information Theory in Scientific Visualization: Part II

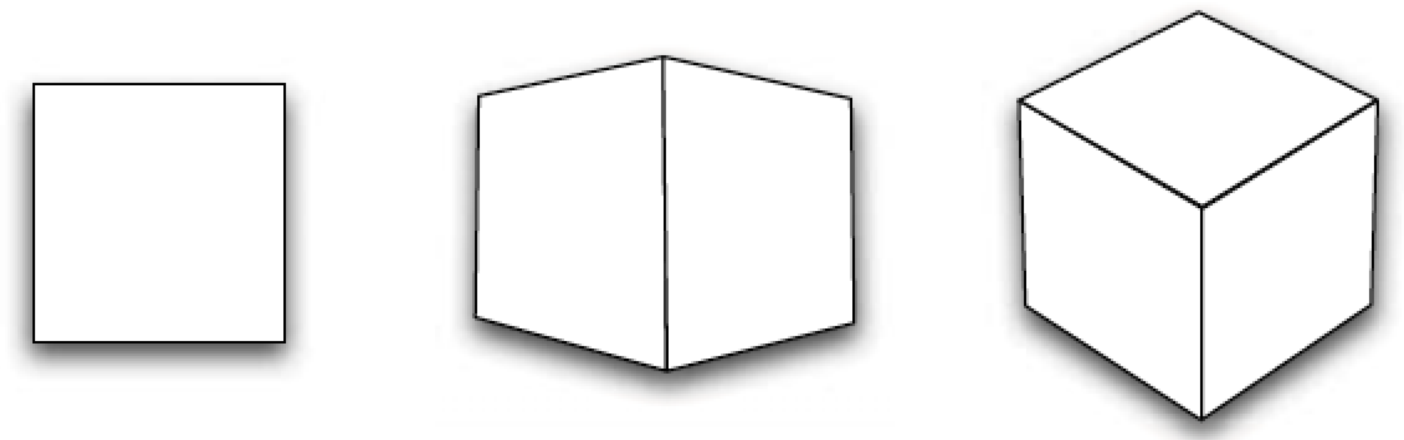
Prof. Ja-Ling Wu
Dept. of CSIE and GINM
National Taiwan University

4. Applications of Information Theory in Scientific Visualization

- *4.1.* ***View Selection for Volumetric Data***
- The goal of **view selection** is to automatically suggest **interesting or optimal viewpoints** that **maximize the amount of information received in the 2D projection of a given 3D dataset**.
- **Good viewpoints** reveal **essential information** about the underlying data. Therefore, presenting them **sooner** to the viewers can improve both the **speed** and **efficiency** of **data understanding**.

- For example, in Figure 4, we show three representative views of a cube with different amounts of information revealed. Clearly, the rightmost one corresponds to the best view which reveals the maximum amount of information about the data by displaying the object in the least uncertain way.
- View selection has its practical value in large-scale data visualization when interactive rendering cannot be achieved.

- Figure 4. Three representative views of a cube showing the increasing amount of information revealed about the object.



- Bordoloi and Shen [7] introduced a solution for view selection for direct volume rendering. They treated the entire volume dataset as a random variable and defined the *visual probability* for a voxel *j* as follows

$$p_j = \frac{1}{\sigma} \cdot \frac{v_j(V)}{W_j}, \quad \text{where } \sigma = \sum_{j=1}^{N} \frac{v_j(V)}{W_j},$$

- where $v_j(V)$ is the *visibility* of voxel *j* at the view V, W_j is the *noteworthiness* of voxel *j* which indicates the significance of its value, and *N* is the total number of voxels in the volume.

- The summation is taken over all voxels in the data. The division by σ is required to make all probabilities add up to unity. The noteworthiness is defined as

  $W_j = \alpha_j I_j = -\alpha_j \log f_j$, where α_j is the opacity of voxel j looked up from the transfer function, I_j is the information carried by voxel j, which can be derived from the frequency of its histogram bin f_j . $-\log f_j$ represents the amount of information associated with voxel j.

- The intuition of their visual probability design can be explained as follows: In **volume rendering**, different voxels contribute differently to the final rendered image.
- The user assigns **high opacity** to voxels that are deemed **more important**. A voxel that is more important, or noteworthy, should be **more visible in the rendering**.

- Conversely, a voxel that is less noteworthy should be less visible.
- Consequently, **the ratio between visibility and noteworthiness should be somewhat even for all voxels to maximize the view entropy**.
- In other words, a **good viewpoint** should strive for a good balance among the visual probabilities of all voxels in the volume so that the information received by the viewer is maximized.

- Takahashi *et al.* [6] considered surface rendering for volumetric data and presented a *viewpoint entropy* measure for iso-surfaces. In this scenario, each iso-surface was treated as a random variable.
- Given an iso-surface I_i, they defined the probability function of a face of the iso-surface as $p_{ij} = A_{ij} / S$, where A_{ij} is the ***visible* area** of the *j*-th face of I_i on the screen and S is the total area of the 2D screen.

- Note that they also included the background area A_{i0} so that the summation of all A_{ij} equals S.

- [6]. Takahashi, S.; Fujishiro, I.; Takeshima, Y.; Nishita, T. A feature-driven approach to locating optimal viewpoints for volume visualization. In Proceedings of IEEE Visualization Conference, Minneapolis, MN, USA, October 2005; pp. 495–502.

- The intuition in their probability function design is that a good viewpoint should **allow each face of the surface to be equally visible**. In this case, the **maximum amount of information about the surface can be received**.
- The entropy of the entire volume takes the average of viewpoint entropies of the extracted iso-surfaces.

- Each contributing iso-surface may carry a **weight** indicating its importance on average. Such a weight can be **derived from the opacity transfer function** (*i.e.*, higher opacity, higher weight). They also extended the same idea to define the viewpoint entropy for interval volumes.

- Ji and Shen [7] took an image-space approach for view selection. Unlike [2], they treated **the rendered image** rather than the volume data as the **random variable**. They considered three aspects of the rendered image, namely, ***opacity***, ***color***, and ***curvature***, to evaluate the **information content** of the image **associated with a given viewpoint**. A good view should **maximize the projection size** and **maintain an even distribution of opacity values**.

[7]. Ji, G.; Shen, H.W. Dynamic view selection for time varying volumes. *IEEE Trans. Vis. Comput. Graph.* 2006,

- It should also **maximize the area of the salient colors** while **maintaining an even distribution of these colors** in the image. Finally, it should allow the viewer to **view the surface curvatures** more easily. Based on this *static* view selection method, they proposed a solution to select ***dynamic* views for time-varying volume data visualization**. Their goal was to maximize the information perceived from the time-varying dataset under the **constraints of smooth view change** and **near-constant speed**.

4.2. Streamline Seeding and Selection

- The concept of entropy can be applied to **detect salient regions** and **generate streamlines for flow visualization**.
- In this case, the **direction of flow** in a **vector field** can be considered as a random variable, and the **distribution of vector directions** indicates the **amount of information** in the vector field.

- Xu *et al.* [8] computed the entropy for every point in **the** vector field by considering its **local neighborhood**. They discretized **vector directions** into a finite number of bins to construct the **histogram**.
- In the resulting entropy field, **high entropy regions** correspond to a **larger degree of variation** in the **vector directions**. These regions are usually near the **critical points** or other important **flow features** such as **separation lines**.

[8]. Xu, L.; Lee, T.Y.; Shen, H.W. An information-theoretic framework for flow visualization. *IEEE Trans.Vis. Comput. Graph.* 2010, *16*, 1216–1224.

- **Streamline seeds** can be placed accordingly to enhance these important features. After a set of streamlines are placed near high entropy regions, to evaluate how well these streamlines represent the underlying flow field, they proposed to **reconstruct an intermediate flow field** and use the **conditional entropy** as the measure.

- In their computation of $H(X|Y)$, X is the **original field** and Y is the **reconstructed field**. The rationale behind it is that **if $H(X|Y)$ is low**, then most of the information in the original field has been **revealed by the reconstructed field**; otherwise, more streamlines need to be seeded.

- The principle for **selecting new seed locations** is that the **higher the conditional entropy** around a spatial point, the more likely the point to be selected as the **next seed**.
- A probability distribution function (PDF) can be constructed to record the **expected probability of dropping a seed** for each point in the domain.
- They distributed the seeds according to the probability distribution function using **importance sampling**.

- Another direction of applying information theory to flow visualization is to **place the focus on traced streamlines** instead of **seed placement**.
- We can apply the **entropy** measure to evaluate the **information content of each individual streamline** by treating **each line as a random variable**. The goal is to prioritize the set of 3D streamlines according to their **entropies** for selective rendering so that **a less cluttered visualization** is presented.

- Furuya and Itoh [9] defined the probability function $p(i)$ as follows

  $p(i) = D_i / L$, where D_i is **the length of the *i*-th streamline segment's** projection on the 2D screen, and ***L* is the total length** of the streamline in the 3D space. The intuition is to **favor streamlines** that have a **nearly equal projected length for all segments**.

- This idea was later adopted by Marchesin *et al.* [10] in their definition of the **linear** and **angular entropies for streamlines**.

- [9]. Furuya, S.; Itoh, T. A streamline selection technique for integrated scalar and vector visualization. In Proceedings of the IEEE Visualization Conference Poster Compendium, Columbus, OH, USA, October 2008.
- [10]. Marchesin, S.; Chen, C.K.; Ho, C.; Ma, K.L. View-dependent streamlines for 3D vector fields. *IEEE Trans. Vis. Comput. Graph.* 2010, *16*, 1578–1586.

4.3. Transfer Function for Multimodal Data

- Haidacher *et al.* [11] proposed an information-based transfer function specification for **multimodal data visualization**. Multimodal visualization **complicates the transfer function design** because **multiple values at every data point need to be considered**. The challenge for multimodal visualization is **how to fuse multiple parameters** in the high-dimensional transfer function space to **enable easy and intuitive transfer function design in the 2D screen space**.

- In this work, the authors considered the **joint occurrence of multiple features** from **one or multiple variables** by utilizing the concept of ***point-wise mutual information*** **(PMI)**.
- The **PMI** of a pair of outcomes $f1$ and $f2$ from **two random variables** describes the **discrepancy** **between the probability of their coincidence** given their **joint distribution** $p(f1, f2)$ versus the probability of their coincidence given only their **individual distributions** $p(f1)$ **and** $p(f2)$, assuming **independence**.

- That is,

  $PMI(f1, f2) = \log p(f1, f2) / p(f1)p(f2)$

- It is clear that $PMI(f1, f2) = 0$ when $p(f1, f2) = p(f1)p(f2)$. This corresponds to the case that the two values are statistically independent from each other. If the pair of values occurs more frequently as one would expect, then $PMI(f1, f2) > 0$. Conversely, if the pair of values occurs less frequently as expected, then $PMI(f1, f2) < 0$.

- The authors leveraged this information as one **additional dimension** to specify the **transfer function** where **high opacity** is assigned to **regions with low PMI**. Thus, statistical features that **only occur in a single variable** can be **separated from** those that are present in both.

- [11]. Haidacher, M.; Bruckner, S.; Kanitsar, A.; Gr.oller, M.E. Information-based transfer functions for multimodal visualization. In Proceedings of Eurographics Workshop on Visual Computing for Biomedicine, Delft, The Netherlands, October 2008; pp. 101–108.

4.4. Selection of Representative Isosurfaces

- **Isosurface rendering** is one of the most popular techniques to **visualize volumetric datasets**. Similar to **isocontours in 2D, isosurfaces in 3D** reveal important **object and/or material boundaries**.
- The key issue is **how to select salient isovalues** such that the surfaces extracted are informative and representative.

[12]. Bruckner, S.; M.oller, T. Isosurface similarity maps. *Comput. Graph. Forum* 2010, *29*, 773–782.

- Conventional solutions made use of **histograms** to depict the **frequency** of isovalues and derived quantities (such as **gradient magnitude**) to suggest interesting isovalues in the plots. Bruckner and M.oller [12] proposed to **evaluate the similarity between iso-surfaces using mutual information**. They produced an *isosurface similarity map* to guide representative isovalue selection.

- Instead of explicit extraction of each individual isosurface for similarity evaluation, they opted to represent individual isosurface implicitly using a **distance transform**. Therefore, in their mutual information $I(X; Y)$ computation, X and Y are actually **the distances from any point in the volume to a pair of isosurfaces *Lp* and *Lq***, respectively.

- The **minimum distance** of a point to the surface was used. The intuition is that **two isosurfaces are similar if their distance distributions are similar and vice versa**.
- To select representative isovalues, they presented an algorithm that automatically **detects coherent structures** (*i.e.*, distinct squares) from the isosurface similarity map and selects the most representative isovalues.

4.5. LOD Selection for Multiresolution Volume Visualization

- Building a **multiresolution data hierarchy** from a **large-scale dataset** allows us to **visualize** the data **at different scales** and **balance image quality** and **computation speed**.
- To construct such a hierarchy, a given volume dataset is first **partitioned into blocks** following either the **bottom-up** or **top-down** strategy.
- A **level-of-detail (LOD)** in the hierarchy consists of a sequence of data blocks at various resolutions.

- The key to **multi-resolution volume visualization** is to select appropriate **LODs** that highlight important features in the data for rendering.
- The goal is to **maximize** the **amount of information** contained in the **image under a certain constraint** about the **computation cost**.

- Wang and Shen [13] proposed to **quantitatively evaluate the *LOD quality*** using the concept of **entropy**.
- They analyzed the LOD quality by investigating the **quality of each individual block** as well as the **relationships** among them. Different blocks may have different ***distortions*** with respect to the original data. They may convey different **optical contents** when the **color** and **opacity transfer function** is applied.

- Furthermore, the sequence of data blocks in the **LOD** are **rendered** to the screen. Different blocks have **different** ***contributions*** to the final image depending on their **projections** and **occlusion relationships**. Therefore, the probability of a multiresolution data block was defined as

$$p_i = \frac{C_i \cdot D_i}{S}, \quad \text{where } S = \sum_{i=1}^{M} C_i \cdot D_i,$$

- where C_i and D_i are the **contribution** and **distortion** of block *i* respectively, *M* is the **total number of blocks** in the data hierarchy.

- The **multiplication of contribution and distortion** in the above equation should be somewhat **even for all blocks** in order to **maximize the LOD entropy**.
- This means that if a data block has **high contribution**, we should **reduce its distortion** by **replacing it** with **its descendant blocks**. Conversely, if neighboring data blocks have **low contribution**, we should **increase** their **distortion** by **replacing** them with their **ancestor blocks**.

[13]. Wang, C.; Shen, H.W. LOD map—A visual interface for navigating multiresolution volume

visualization. *IEEE Trans. Vis. Comput. Graph.* 2006.

- Note that for any LOD, **it is impossible** for **all the data blocks** in the hierarchy to have the **equal probability**. This is because **an LOD constitutes a *cut* in the data hierarchy** and thus **not all of the data blocks** can be **selected**. Any block which is not included in the LOD receives zero probability and does not contribute to the entropy.

- Ideally, since a **higher entropy** indicates a **better LOD quality**, the **best LOD** (with the **highest information content**) could be achieved when we **select all the leaf nodes** in the **data hierarchy**. However, this requires rendering the volume data at the **original resolution**, and defeats the purpose of multiresolution rendering. In practice, a meaningful goal is to **find the best LOD under some constraint**, such as a certain **block budget**, which is usually much smaller than *M*.

- Accordingly, the quality of an LOD could be improved by **splitting data blocks with large distortion and high contribution**, and **joining those blocks with small distortion and low contribution**.
- The **split operation** aims at i**ncreasing the entropy** with a more **balanced probability distribution**. The **join operation** is to **offset the increase in block number** and **keep it under the budget**.

4.6. Time-varying and Multivariate Data Analysis

- **Time-varying** and **multivariate data analysis** and **visualization** has received increasing attention in recent years.
- **Identifying important regions** in the data enables **effective data reduction**, **viewing**, and **understanding**, which provides a **scalable solution** to **handle large-scale data**.

- Janicke *et al.* [14] introduced an approach to **detect importance regions** for **multi-field data** by extending the concept of ***local statistical complexity*** (LSC) from **finite state cellular automata** to **discretized multi-fields**.
- They defined **past and future light-cones** (*i.e.*, **influence regions**) for all grid points, which are used to estimate **conditional distributions** and calculate the LSC.

- Specifically, the **LSC** at a grid point *p* was defined as

$$C(p) \equiv I[\varepsilon(l^-(p)); l^-(p)], \quad \text{where } \varepsilon(l^-) = \{\lambda | P(l^+|\lambda) = P(l^+|l^-)\},$$

- where $l^-(p)$ and $l^+(p)$ are the **configurations of the field** in the past and future cones respectively, ***ε*** is the ***minimum sufficient statistic*** which maps **past configurations** to their **equivalence classes** (*i.e.*, classes with the same conditional distribution $P(l^+ / l^-)$), $\varepsilon(l^-(p))$ are the ***causal states*** of the system which **predict the same possible futures with the same possibilities**.

[14]. J.anicke, H.; Wiebel, A.; Scheuermann, G.; Kollmann, W. Multifield visualization using local statistical complexity. *IEEE Trans. Vis. Comput. Graph.* 2007.

- Intuitively, **mutual information** $I[\varepsilon(l^{-}); l^{-}]$ indicates the **minimum amount of information** of a **past light-cone** needed to **determine its causal state**.
- Thus, the LSC tells **how complex** it is around the **past configuration centered at point *p***. The **higher** the $C(p)$, the **more complex** the **local region around *p***.

4.7. Information Channel between Objects and Viewpoints

- Compared to previously described visualization examples, the work by Viola *et al.* [15] on **importance-driven focus of attention** is unique in the sense that they **built an information channel** in terms of ***visibility* between objects and viewpoints**.

- Previously Sbert *et al.* [16] showed that for **polygonal data**, the **viewpoint entropy** [17] is very **sensitive to the discretization** of the objects.
- Viola *et al.* built the **information channel between two random variables** (the input, *i.e.*, **viewpoints,** and the output, *i.e.*, **objects**) by computing a **probability matrix** which determines **the output distribution given the input**.

- They defined a new measure, called the ***viewpoint mutual information* (VMI)**, which is better than the viewpoint entropy due to its **robustness** to deal with **any type of discretization or resolution** of the volumetric dataset.

- [15]. Viola, I.; Feixas, M.; Sbert M.; Groller, M.E. Importance-driven focus of attention. *IEEE Trans. Vis. Comput. Graph.* 2006, *12*, 933–940.
- [16]. Sbert M.; Plemenos, D.; Feixas, M.; Gonzalez, F. Viewpoint quality: Measures and applications. Proceedings of Eurographics Workshop on Computational Aesthetics in Graphics, Visualization and Imaging, Girona, Spain, May 2005; pp. 185–192.
- [17]. Vazquez, P.P.; Feixas, M.; Sbert, M.; Heidrich, W. Viewpoint selection using viewpoint entropy. In Proceedings of Vision, Modeling, and Visualization Conference, Stuttgart, Germany, November 2001; pp. 273–280.

- The **mutual information** between a set of **viewpoints *V*** and a set of **objects *O*** is defined as

$$I(V;O) = \sum_v p(v) \sum_o p(o|v) \log \frac{p(o|v)}{p(o)} = \frac{1}{|V|} \sum_v I(v;O),$$

where

$$I(v;O) = \sum_o p(o|v) \log \frac{p(o|v)}{p(o)}.$$

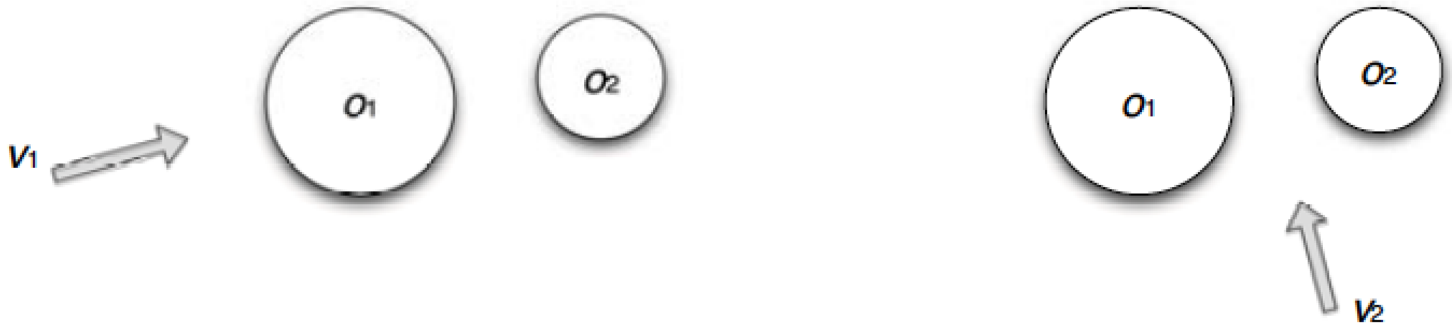
- In the above equations, $p(v) = 1/|V|$, *i.e.*, each viewpoint has an equal probability. $p(o|v)$ is the normalized visibility of object o from viewpoint v and $\sum_o p(o|v) = 1.0$. $p(o)$ is the average visibility of object o obtained from the set of viewpoints V, *i.e.*,

$$p(o) = \sum_v p(v)p(o|v) = \frac{1}{|V|}\sum_v p(o|v).$$

- ***I*(*v*;*O*)** is the VMI, which indicates **the degree of dependence or correlation** between the dataset ***O*** and viewpoint ***v***. We sketch two examples to illustrate the intuition of this VMI measure. In the left image of Figure 5, the viewpoint v_1 and the set of objects *O* are highly coupled, *i.e.*, the **average visibility** of o_1 and o_2 is **low** due to the **occlusion** of o_2 by o_1.

- This implies that ***$I(v_1;O)$* has a high value** which corresponds to a **low quality viewpoint**. In the right image of Figure 5, the viewpoint v_2 and the set of objects O are more independent, *i.e.*, the two objects o_1 and o_2 are **equally visible** from v_2 without occluding each other.
- This implies that ***$I(v_2;O)$* has a low value** which corresponds to a **high quality viewpoint**. The ***best* viewpoint** is achieved when ***$I(v;O)$* is *minimized***.

- Figure 5. Illustration of the **viewpoint mutual information**. Left: a low quality viewpoint indicating a highly dependent view between the viewpoint v_1 and the set of objects $O = \{o_1, o_2\}$. Right: a high quality viewpoint indicating a more independent view between the viewpoint v_2 and the set of objects O.



- Leveraging Bayes' theorem, *i.e.*, *p*(*v*)*p*(*o*/*v*) = *p*(*o*)*p*(*v*/*o*), Ruiz *et al.* [18] reversed the information channel proposed by Viola *et al.* [15].

$$I(O;V) = I(V;O) = \sum_v p(v) \sum_o p(o|v) \log \frac{p(o|v)}{p(o)}$$

$$= \sum_o p(o) \sum_v p(v|o) \log \frac{p(v|o)}{p(v)} = \sum_o p(o) I(o;V),$$

where

$$I(o;V) = \sum_v p(v|o) \log \frac{p(v|o)}{p(v)}.$$

- In their context, *o* represents each individual voxel in the volume. Therefore, they defined $I(o; V)$ as the *voxel mutual information*, which was utilized in various visualization applications such as volume illustration and viewpoint selection.

- [18]. Ruiz, M.; Boada, I.; Feixas, M.; Sbert, M. Viewpoint information channel for illustrative volume rendering. *Comput. Graph.* 2010, *34*, 351–360.