An Entropy-Based Approach for Identifying User-Preferred Camera Positions

Nicole Marsaglia* University of Oregon
Yuya Kawakami University of Oregon
Samuel D. Schwartz University of Oregon
Stefan Fields University of Oregon
Hank Childs University of Oregon

ABSTRACT

Viewpoint Quality (VQ) metrics have the potential to predict human preferences for camera placement. With this study, we introduce new VQ metrics that incorporate entropy, and explore how they can be used in combination. Our evaluation involves three phases: (1) creating a database of isosurface imagery from ten large, scientific data sets, (2) conducting a user study with approximately 30 large data visualization experts who provided over 1000 responses, and (3) analyzing how our entropy-based VQ metrics compared with existing VQ metrics in predicting expert preference. In terms of findings, we find that our entropy-based metrics are able to predict expert preferences 68% of the time, while existing VQ metrics perform much worse (52%). This finding, while valuable on its own, also opens the door for future work on in situ camera placement. Finally, as another important contribution, this work has the most extensive evaluation to date of existing VQ metrics to predict expert preference for visualizations of large, scientific data sets.

Index Terms: Viewpoint Quality Metrics—View-Dependent Visualization—Automating Visualization—

1 INTRODUCTION

Camera placement is a critical task for scientific visualization. In a typical post hoc setting, the camera placement process typically starts with a default camera position (e.g., zoomed out with the camera translated down the Z-axis and pointed at the center of the data set) and a domain scientist uses a mouse to modify the camera location to a position or positions that increase their insight. That said, this mode of interaction is not the only approach. Alternatively, the camera can be set via an automated approach. If the automated approach can select insightful views, then there are multiple applications. One application is in the post hoc context, whether for setting the initial camera position or for suggesting insightful views to users. Another important application is in the in situ setting, which frequently has no human in the loop. In this setting, automation is an important approach, i.e., not only having in situ visualization routines perform tasks like rendering, but also having them perform tasks like determining the camera positions for those renderings.

A significant number of research investigations have considered using viewpoint quality (VQ) metrics to evaluate the quality of a camera placement. These metrics are designed to produce "better" views as the metric values increase. That said, the evaluation of these metrics has been limited to non-scientific data sets and the metrics have not considered in situ constraints. With this work, we fill this gap by conducting a user study of large data visualization practitioners using isosurface imagery. Further, while we do not apply our results in situ, we do constrain the metrics considered to those that can be efficiently calculated in such a setting.

As a separate contribution, we introduce new VQ metrics that are based on entropy. We introduce three such metrics, one that measures data entropy, one that measures depth entropy, and one that measures shading entropy. Many previous research efforts have considered entropy for optimizing visualization parameters; the novelty in our work is in the specific form for this specific problem. We also introduce a mechanism for combining metrics. Our findings show that these metrics perform better than comparator VQ metrics.

Summarizing, this work has two significant contributions:

- We introduce new VQ metrics that are appropriate for an in situ setting and demonstrate that these metrics perform better than existing metrics.
- We conduct the first ever user study devoted to large, scientific visualizations, and use the results to evaluate the efficacy of ten VQ metrics.

2 RELATED WORK

Foundational research on user preferences for camera position has occurred outside of the graphics and visualization communities. Tarr and Kriegman [40] conducted psychoanalysis experiments investigating the influence of viewpoint on user preference and found that, for many models, there exists a small number of views that are preferred by most people. Further, numerous user studies have shown that users prefer an image with a three-quarter or "canonical" view [5, 27]. According to Blanz et al. [5], canonical views are stable and expose as many salient and important features as possible. That said, Kamada et al. [19] considered an image good if it minimizes the number of degenerate views. But while these views have been proven to be visually pleasing, they provide no guarantee of scientific merit or importance.

Within the graphics community, there have been multiple efforts at defining viewpoint quality (VQ) metrics. A VQ metric takes a data set and a camera position as input and returns a score (floating-point value) as output. This score reflects the metric's assessment of the quality of the camera position for the given data set. Multiple works have surveyed viewpoint quality metrics [14, 31, 34], with the most recent by Bonaventura et al. [7]. The Bonaventura survey organized VQ metrics into five categories based on what aspects of the data are utilized: area [6, 15, 29, 30, 32, 33, 42], depth [34, 37], silhouette [26, 31, 34, 43], image stability [4, 8, 9, 15, 24, 42], and surface curvature [15, 17, 22, 26, 31, 36].

Further, some proposed methods in the graphics community have incorporated entropy. Vázquez et al. [42] calculated the entropy of projected areas of the visible geometry from a given viewpoint. Next, Gumhold [16] applied entropy to the Phong-Blinn illumination and chose a light source that maximized this information for a given data model. Later, Page et al. [26] proposed calculating the entropy of a data model's silhouette, or outline, as well as calculating the entropy of the data model's surface curvature, but it was Polonsky et al. [31] who first applied these entropy calculations as VQ metrics.

Within the visualization community, there have been multiple efforts at locating good camera positions. Bordoloi and Shen [8] argued that the "best image" depends on context. They have two guidelines for determining a "good image" for volume rendering: viewpoints that display voxels with high noteworthiness factors and viewpoints where the projection contains a high amount of information. Work from Correa and Ma [12] also involved transfer

*e-mail: marsagli@uoregon.edu

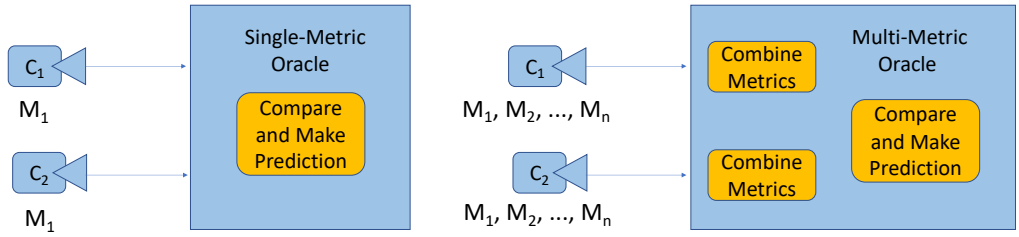

Figure 1: A single-metric oracle (Left) and a multi-metric oracle (Right). Single-metric oracle: For two cameras, C_1 and C_2, and a single metric, M_1, the oracle will compare each camera's respective metric score and determine the best viewpoint. Multi-metric oracle: For two cameras, C_1 and C_2, and n metrics, $M_1,\ldots,M_n$, the oracle will combine each camera's metric scores, compare the combined scores, and determine the best viewpoint.

functions, allowing users to decide importance metrics that will be used to create a transfer function that highlights intervals of interest. Viola et al. [45] also used importance features to determine a transfer function, and then utilized Viewpoint Mutual Information (VMI) to determine the best viewpoint [44]. Finally, work from Yao and Wang [46] incorporated AI to determine both the transfer function and the best viewpoint.

The visualization community has also considered using entropy to improve rendering. Takahashi et al. [39] considered volume rendering use cases, focusing on projections of subregions. Specifically, they considered the entropy of individual weighted and unweighted isosurfaces from different camera positions. The former has trouble accounting for occlusions, whereas the latter requires a transfer function. Takahashi et al. [38] then developed two new metrics that summate the entropy of individual weighted and unweighted interval volumes. Other notable works include a method by Naraoka et al. [25] that optimized light sources for volume rendering via illumination entropy and a method by Lee et al. [23] that used the entropy of a vector field for flow visualization, specifically for choosing seed placements and for choosing optimal viewpoints.

Evaluation of user preference for camera placement has primarily occurred within the graphics community. Dutağaci et al. [14] created a benchmark for testing new viewpoint quality metrics against the preference of 26 human subjects for 68 recognizable 3D objects. To collect this data, they asked their subjects to select the most informative view of each object. A distance measure was then used to determine how close a proposed metric was to selecting a user-preferred view. Next, where the Dutağaci et al. work evaluated seven metrics using their benchmark, a survey by Bonaventura et al. [7] evaluated all 22 metrics against the benchmark. Finally, Secord et al. [34] and Polonsky et al. [31] both described and evaluated metrics and concluded that no current metric was sufficient for consistently finding the best viewpoint, and, Polonsky et al. speculated finding the best viewpoint may require combining several metrics. Secord et al. pursued this direction, combining metrics based on different data attributes, such as surface visibility, silhouette length, projected area, and maximum depth.

Our research differs from these previous works. Many of the previous works lacked user evaluation, instead proposing a metric and then demonstrating they could maximize that metric. In our case, evaluation is done through a user study that includes the perspectives of many visualization practitioners and many data sets. Other VQ metric works have included evaluation, but have not considered scientific data sets. Our efforts fill this gap, i.e., a user study on scientific data sets. It also contributes new VQ metrics that are more effective for our data sets than previous VQ metrics.

3 Our Method

This section details our method for constructing an oracle that can use VQ metrics to predict user preference. There are two main concepts in this section: (1) how to construct an oracle from VQ metrics? and (2) which VQ metrics do we incorporate into oracles? The first concept is discussed in Section 3.1. The second concept is discussed in two parts: Section 3.2 discusses new entropy-based VQ metrics that we introduce in this work and Section 3.3 discusses existing VQ metrics that we use as comparators.

3.1 Constructing Oracles from VQ Metrics

A VQ-based oracle uses VQ metric values to predict which camera a user would prefer. In an in situ setting, they could be used to automate camera placement: evaluating camera positions and ultimately selecting the one thought to best match user preference. However, in this study, oracles are used in a more limited way: to evaluate which VQ metric best matches user preference. Further, in this study, an oracle considers two camera positions and attempts to predict the user's preference between the two. That said, oracles could trivially be expanded to deal with more than two camera positions.

For the version of oracle we consider, the only inputs are VQ metrics. They do not have access to image data, information about the camera position, or the geometry being rendered. For example, one of the VQ metrics we consider is "Visibility Ratio" (described in Section 3.3.4) and one of the oracles we construct attempts to predict user preference using only Visibility Ratio. In this case, the only input to the oracle would be the values for Visibility Ratio for the two camera positions.

As shown in Figure 1, oracles can operate using only a single VQ metric or multiple VQ metrics. The key distinction between them is that single-metric oracles do not need to combine metrics.

3.1.1 Single-Metric Oracles

A single-metric oracle is an oracle that uses only one metric to make decisions. We created 10 single-metric oracles, one each for our three new entropy metrics (Section 3.2) and one each for the seven existing metrics (Section 3.3). If a metric M produces score $M(C_1)$ for camera position C_1 and score $M(C_2)$ for camera position C_2 and if $M(C_1) > M(C_2)$, then the oracle would select C_1. For all of the metrics, "bigger is better," so their corresponding oracles choose the higher values. Further, it was not necessary to decide how to deal with equal metric values, since this did not occur in our experiments.

3.1.2 Multi-Metric Oracles

An oracle that uses multiple metrics has the potential to make better decisions by drawing on different types of information. Given a multi-metric oracle that uses n metrics, $M_1,\ldots,M_n$, these metrics produce tuple $(M_1(C_1)\ldots,M_n(C_1))$ for camera position C_1 and tuple $(M_1(C_2),\ldots,M_n(C_2))$ for camera position C_2. That said, these tuples

are not immediately useful, as the metrics produce values with disparate ranges and a variety of units — combining the metrics to make a binary decision is a a fundamental issue with the multi-metric approach. Our solution is to try three different methods for combining metric scores and evaluate all three. That said, looking ahead to results, all three combination methods produced similar findings. The three methods we consider for combining metrics are:

- **NORM**: Normalizing the scores based on minimum and maximum values and adding the normalized scores together.
- **TIER**: Clustering the scores into tiers using an automated method (Jenks natural breaks optimization [18]) and then adding the tiers together.
- **NONE**: Adding the raw scores together.

The **NONE** approach is not appropriate in almost all cases, but it is appropriate in the case of adding together three entropy scores.

Once the scores were combined, they were compared, and the oracle selected the C_i with the highest value. The **TIER** approach did have ties in some cases, and in these cases a random camera was chosen.

3.2 New VQ Metrics

Each of our new metrics utilizes Shannon Entropy [4], which calculates the average level of information. Given a discrete random variable X, with the possible outcomes $x_1, \ldots, x_n$, occurring with the respective probabilities $P(x_1), \ldots, P(x_n)$ the entropy of X is defined as:

$$H(X) = -\sum_{i=1}^{n} P(x_i) \cdot \log(P(x_i))$$

The higher the entropy the more information that is present.

Entropy-based VQ metrics can be constructed by placing fields on images in addition to the typical colors. For example, graphics libraries often produce depth information for the Z-buffer algorithm, and this data augments the image. This depth information can then be used as the discrete random variable for the Shannon entropy calculation, i.e., construct a discrete random variable made up of the depth information for every pixel where data appears and then calculate Shannon entropy on that random variable. Further, maximizing the score of a given entropy-based VQ metric equates to maximizing the information present in the image, at least with respect to its type of data (for example, maximizing entropy in depth information).

We pursued three types of data entropy-based VQ metrics:

- Field Data: the visible data of some user specified field
- Depth Data: the distance from the camera to the visible field data
- Shading Data: the shading coefficients for the visible geometry of the data

These metrics correspond to readily available quantities during the rendering process (field value, depth value, and normal value, which becomes a shading value).

In all cases, we considered the "visible" data (i.e., visible field data, visible depth data, or visible shading data). This means that a scene is rendered, an image is produced, and the data from that image (field, depth, shading) is extracted. If an image has N pixels, and if M pixels have no data occupying that pixel (i.e., background color), then the visible data comes from the $N-M$ pixels that do overlap with the geometry. Considering the example of shading data on an image of 10 pixels ($N == 10$), if 3 pixels have no data ($M == 3$), then the visible shading data would be the set of shading values from the remaining 7 pixels ($N-M$), e.g., $\{0.3, 0.2, 0.6, 1.0, 1.0, 0.8, 0.2\}$.

3.2.1 Data Entropy

Data Entropy calculates the entropy of the visible field data from a given viewpoint v. Given field data F, let $F(v)$ be the visible field data for some viewpoint v with elements $f_1, \ldots, f_n$, then Data Entropy is defined as:

$$H(F(v)) = -\sum_{i=1}^{n} P(f_i) \cdot \log(P(f_i))$$

Past research failed to develop this metric because they were primarily focused on developing viewpoint quality metrics for 3D objects that only have geometric data and lack any field data, unlike scientific data. And while Data Entropy can be applied to any field mesh, it is best if the geometry is unstructured and produces amorphous shapes, as opposed to a three-dimensional rectilinear mesh.

3.2.2 Depth Entropy

Depth Entropy calculates the entropy of the distances from the camera to the visible field data from a given viewpoint v. Let $D(v)$ be the set of distances from the camera to the visible field data for some viewpoint v with elements $d_1, \ldots, d_n$, then Depth Entropy is defined as:

$$H(D(v)) = -\sum_{i=1}^{n} P(d_i) \cdot \log(P(d_i))$$

Depth entropy is similar to Secord et al.'s [34] metric, Depth Distribution, which deals with the normalized histogram of depth bins. Finally, this metric is readily applicable to surface data, which fits our isosurface-centric study. Volume rendering would require extending this metric, e.g., adapting for regions of high opacity or when the opacity along a ray hits a threshold.

3.2.3 Shading Entropy

Shading Entropy calculates the entropy of the visible shading coefficients. This metric determines the shading coefficient for each visible triangle and then calculates the entropy.

Let $G(v)$ be the set of visible shading coefficients from a viewpoint v, with elements $g_1, \ldots, g_n$, then Shading Entropy is defined as:

$$H(G(v)) = -\sum_{i=1}^{n} P(g_i) \cdot \log(P(g_i))$$

We used flat shading in our calculations, since that was straightforward in our infrastructure, but we note that vertex shading is also possible. Additionally, our infrastructure uses a "miner's light" that is always located above the camera. And while this work is the first to use shading entropy for viewpoint selection, this metric was first proposed by Gumhold [16] who used shading entropy to determine optimal placement for light sources.

3.3 Comparators: Existing VQ Metrics

We consider seven existing VQ metrics as comparators. There are other VQ metrics beyond these seven, but we are only interested in those that can be extended to run both in an in situ setting and in a distributed-memory parallel setting. In all, we only considered a VQ metric if it met three requirements:

- The metric should have a small memory footprint.
- The metric should have a fast execution time.
- The metric should require minimal communication.

The remainder of this section describes the seven metrics, first defining how the metric works and then discussing its merits. All descriptions use the notation summarized in Table 1.

Notation	Definition
x	data value
X	set of data values
$p(x)$	probability of x
z	polygon
Z	set of polygons
v	viewpoint
V	set of viewpoints
$a_z(v)$	projected area of polygon z from viewpoint v
$a_t(v)$	projected area of the model from viewpoint v
$vis_z(v)$	visibility of polygon z from viewpoint v (0 or 1)
N	number of polygons
R	number of pixels of the projected image
A_z	area of polygon z
A_t	total area of the model
$p(z\|v)$	conditional probability of z given v
$p(z)$	probability of z
$H(Z)$	entropy of the set of polygons
$H(Z\|v)$	conditional entropy of the set of polygons given viewpoint v
F	set of field data
$F(v)$	set of visible field data from viewpoint v
$D(v)$	set of distances from the camera to the visible field data from viewpoint v
$S(v)$	set of shading data from viewpoint v

Table 1: Notation used in metric descriptions. This notation was developed by Bonaventura et al. [7].

3.3.1 Number of Visible Triangles

This metric, developed by Plemenos [29] and then expanded upon by Plemonos and Benayada [30], is based on the number of visible triangles from some viewpoint. The best viewpoint will be the one with the highest number of visible triangles. Formally:

$$VQ_1(v) = \sum_{z \in Z} vis_v(z).$$

In the worst case, this metric favors quantity over quality and may choose viewpoints that contain a lot of polygons but little content.

3.3.2 Projected Area

This metric, developed by Plemenos and Benayada [30], favors viewpoints that show the most projected area of the data model. This metric simply sums the visible area of the data's geometry. Formally:

$$VQ_2(v) = a_t(v).$$

In the worst case, this metric could select an image that favors one large polygon. Further, maximizing the projected area of the data could also maximize the number of occlusions.

3.3.3 Plemenos and Benayada

This metric, from Plemenos and Benayada [30], is a combination of their first two metrics and is defined as follows:

$$VQ_3(v) = \frac{\sum_{z \in Z} \lceil \frac{a_z(v)}{a_z(v)+1} \rceil}{N} + \frac{\sum_{z \in Z} a_z(v)}{R},$$

Correcting the downside from the first two metrics, Plemenos and Benayada developed a metric that maximizes the number of visible triangles as well as the resolution of the rendered image. While this metric is an improvement, it is susceptible to favoring viewpoints with large occlusions since this metric can be dominated by visible surface area.

3.3.4 Visibility Ratio

This final metric from Plemenos and Benayada [30] is the ratio of the real visible surface area over the total real surface area (i.e. the areas in World Space) and is defined as follows:

$$VQ_4(v) = \frac{\sum_{z \in Z} vis_v(z) A_z}{A_t}$$

Note that this metric uses the world space geometry rather than device space geometry. This metric has similar advantages and disadvantages as the second metric, VQ_2.

3.3.5 Viewpoint Entropy

This metric was first applied to viewpoint selection by Vázquez et al. [42]. Their metric alters Shannon Entropy [4, 13] to take into account the projected area of the scene when centered at a particular viewpoint. Viewpoint Entropy is defined as follows:

$$VQ_5(v) = -\sum_{i=0}^{N} \frac{a_z(v)}{a_t(v)} \log \frac{a_z(v)}{a_t(v)}.$$

The ratio $\frac{a_z(v)}{a_t(v)}$ represents the proportion of the projected area of each polygon. This ratio is also proportional to the cosine of the angle between the normal of the projected polygon $a_z(v)$ and the camera angle. Additionally, this ratio is inversely proportional to the squared distance from the camera to polygon. This means that $\frac{a_z(v)}{a_t(v)}$ will be higher when the polygon is seen from a better angle and at a closer distance. This metric will work best on data sets with varying polygonal size, since larger polygons are penalized in comparison to smaller polygons. An important drawback of this metric is that it will go towards infinity with finer mesh resolutions.

3.3.6 Viewpoint Kullback-Leibler distance (VKL)

Developed by Sbert et al. [33], this metric measures the Kullback-Leibler distance between the normalized distribution of the projected areas of polygons from a given viewpoint and the normalized distribution of the real areas of polygons. It is defined as:

$$VQ_6(v) = \sum_{z \in Z} \frac{a_z(v)}{a_t(v)} \log \frac{\frac{a_z(v)}{a_t(v)}}{\frac{A_z}{A_t}}.$$

The best viewpoint, which corresponds to the minimum value, is achieved when the normalized distribution of the projected areas is equal to the normalized distribution of real areas. In order to make the metrics all follow a "bigger is better" pattern, we multiply this value by -1.

3.3.7 Maximum Depth

This metric was defined by Stoev and Strasser [37], but was applied by Secord et al. [34] as a VQ metric. This metric is defined as follows:

$$VQ_7(v) = depth(v),$$

where $depth(v)$ is the maximum depth of the model from some viewpoint v. Depth can be a useful metric for terrain data sets. Terrain data sets are often viewed from above when information is maximized, making the data appear flat, thus it is important to take depth into consideration.

4 Corpus for Comparing Viewpoints

This section describes our data corpus for evaluating our method. The corpus is made up of multiple elements. First, the corpus contains images and meta-data about these images. This aspect of the corpus is discussed in Section 4.1. Second, the corpus contains results from a user survey on preferred images. This aspect of the corpus is discussed in Section 4.2.

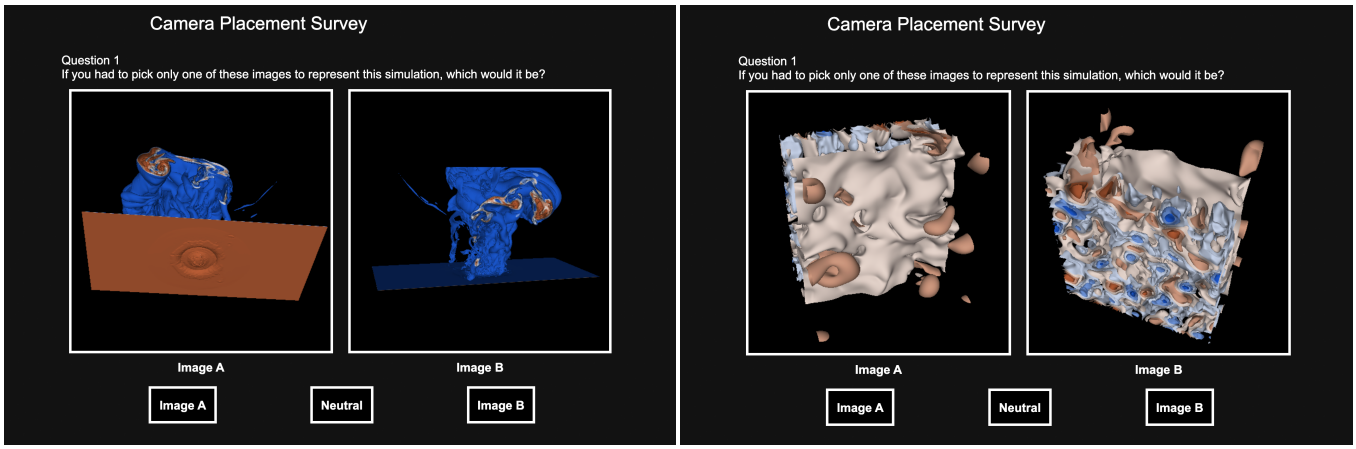

Figure 2: Examples of two questions from the user survey. The left image depicts a question between two viewpoints from the Asteroid data set. The right image depicts a question between two viewpoints from the S3D-UVEL data set. Users can select the image they feel is most representative of the simulation, or they can select neutral if they have no preference.

4.1 Generating a Database of Images

Our corpus considers multiple camera positions for multiple scientific data sets. For each (camera, data set) pair, the corpus contains an image and all VQ metric scores for that image. In all cases, the visualization was of a multi-level isosurface. The remainder of this section describes more detail on the data sets used, the selection of isovalues, and how cameras were placed.

4.1.1 Data Sets

A gap in prior research is the lack of application to scientific data sets. To fill this gap, we chose ten large-scale scientific data sets, drawing the IEEE Visualization Conference's Scientific Visualization Contest and from data sets from the Exascale Computing Project from the United States' Department of Energy. One of our primary goals in selecting these data sets was to consider diverse application domains and diverse imagery, so our results would be (as much as possible) applicable to a larger proportion of scientific data sets.

Four data sets were from the Scientific Visualization Contest:

- Asteroid: A data set of a deep water impact of an asteroid [28].
- Fluid Dynamics: A fluid dynamics data set that models a cylindrical flow of water [20].
- Hurricane: A weather data set of Hurricane Isabel [1].
- Mantle: An earth sciences data set that models the Earth's mantle [35].

Six data sets were from the Exascale Computing Project:

- Constit: A material sciences data set that probes the deformation response of polycrystalline materials [10].
- ExaAm Truchas: A materials science data set that looks at effects within micro-structures of Additive Manufacturing (AM) [3].
- ExaSky Nyx: A cosmological data set that looks at gas dynamics [2].
- Miranda: A hydrodynamics data set of large-scale turbulence [11].
- S3D-N2: A combustion data set of field data N2 [41].
- S3D-UVEL: A combustion data set of field data U Velocity [41].

For each of the chosen data sets we selected a single timeslice we felt was representative of the simulation.

4.1.2 Choosing Isovalues

Each of the data sets are three-dimensional and volumetric, and to each we applied isosurfacing as our visualization operation. Six isovalues were selected, specific to each data set. Our process for choosing isovalues was as follows. Initially, default isovalues were chosen and the data was rendered. If the resulting image is considered "good," we keep those values. Otherwise, we explored the data set to choose isovalues that are "good." "Good" was taken to mean "not bad," as in: the isosurfaces occupied a significant portion of the possible volume and (within reason) the isosurfaces were not bunched together with little separation.

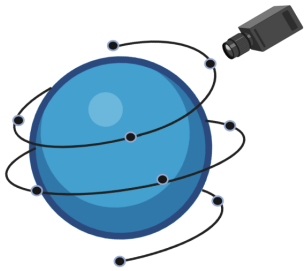

Figure 3: Example of using a Fibonacci Lattice to equally space cameras around a data set. This research used this method determine the ten camera placements for the user survey data sets.

4.1.3 Camera Placement

For each data set, we rendered ten images from ten different viewpoints that can be seen in Figure 4. The camera placements were chosen using Fibonacci's Lattice, a formula that equally spaces points around a sphere, as shown in Figure 3.

We experimented with many camera placement techniques and also with the number of camera positions to include in the survey. We felt the Lattice approach and this number of views provided a nice compromise between two factors. First, we felt the camera positions covered the space of all possible camera positions well — every feature was covered by at least one image. Second, we felt that a small number of camera positions was beneficial, so we could investigate issues like participant disagreement; if the number of views is very large, then it becomes less likely to get two different participants considering the same pair.

4.2 User Study

Our user study collected participant preferences on camera position. The participants in the survey were attendees of the 2021 United States' Department of Energy's Computer Graphics Forum, which is made up large data visualization practitioners across a wide variety

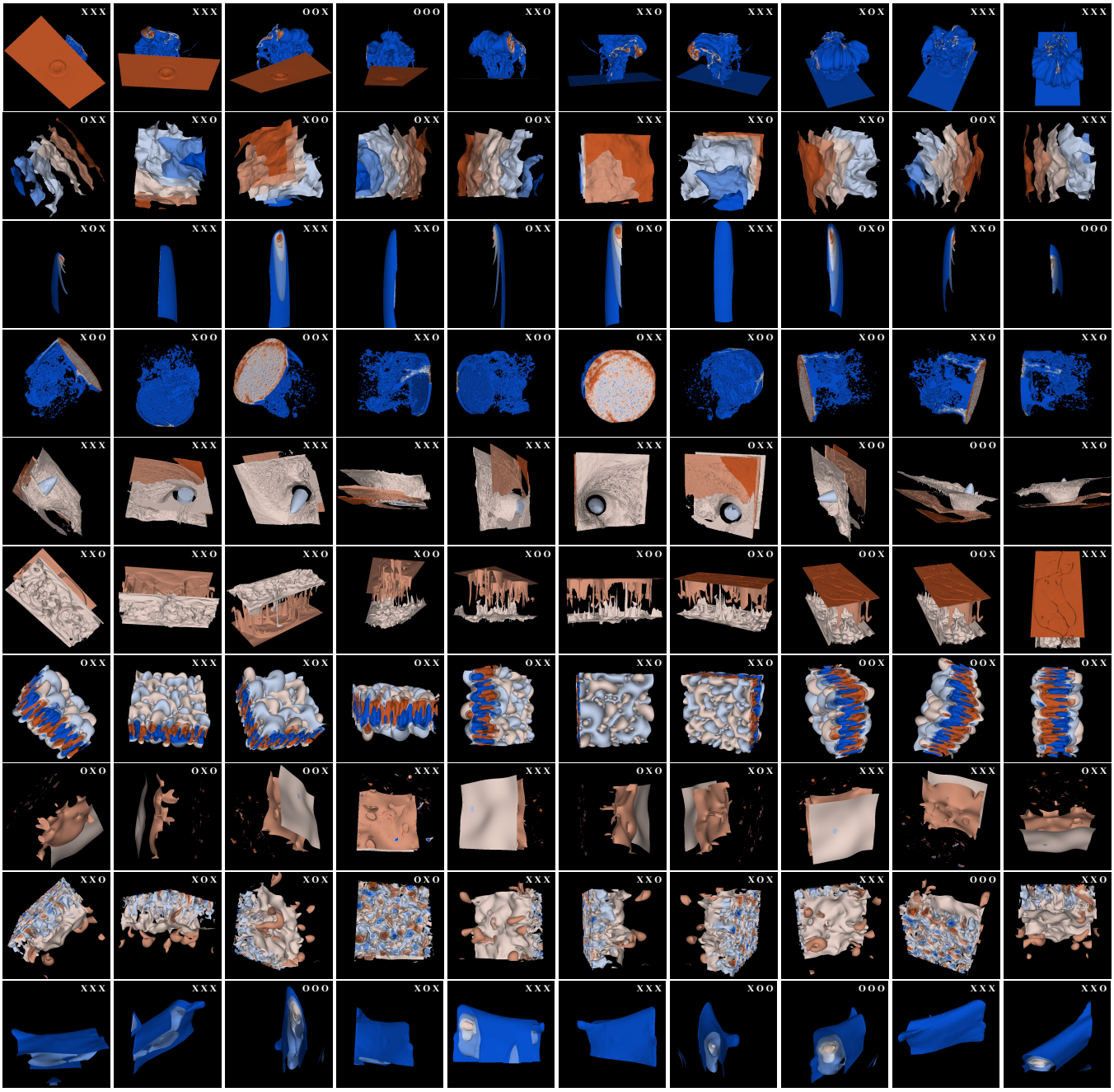

Figure 4: The image set for our corpus. Each data set was transformed into a multi-level isosurface, using six isovalues unique to each data set. Then, using the Fibonacci Lattice, each data set was rendered from ten equally-spaced camera positions around the data set. Each row of images is from the same scientific data set. Each column of images is using the same camera placement. The data sets are, from top to bottom: Asteroid, Constit, ExaAm Truchas, Fluid Dynamics, Hurricane, Mantle, Miranda, S3D-N2, S3D-UVEL, ExaSky Nyx. In the top right corner of each image is an annotation representing the data entropy score, depth entropy score, and shading entropy score, respectively, for each viewpoint. For each data set and metric, the scores were normalized to be between [0,1]. An **O** means the viewpoint has a metric score in the top 20% among the ten images. An **X** means the viewpoint has a metric score in the bottom 80% among the ten images.

of simulation domains. Participants were instructed to make their decision around one central question: "if you had to pick only of these images to represent this simulation, which would it be?"

To take the survey, participants accessed a website where they were presented with a sequence of questions. Each question was composed of two images from the same data set, as shown in Figure 2. Participants were asked to spend 10 minutes answering questions, though they could stop whenever they wanted. Participants answered the question by selecting the image they felt was most representative, or neutral. Having answered, a new pair of images is generated for that user to compare. The questions are randomly generated on demand for each user. To generate a question, first the data set was randomly chosen, then the two images to be compared ware randomly chosen, making sure to not to select the same view-

point for both images, and making sure not to repeat any pairs of images that user has already seen. It is believed that approximately 30 visualization practitioners participated in the survey. (If a participant closed their web browser and restarted the survey, then they would appear as a new participant, making an exact count difficult.)

The survey resulted in 1266 responses, although 170 of these responses indicated the participant had no preference for one image over another. We discarded these "neutral" responses, resulting in 1096 entries where the participant had a preference. Each of these 1096 entries is a tuple of the form (D, C_i, C_j, R) where D is one of the ten data sets, C_i and C_j are camera positions, and R is the participant preference (i.e., $R = C_i$ or $R = C_j$).

With respect to metric information, we calculated the ten VQ metric values for each of the 100 combinations of camera position and data set, and these 1000 values complemented our corpus. The values were calculated using Ascent [21], which has implementations to calculate all ten VQ metrics (although some implementations have not yet been merged to the main Ascent repository). For the entropy calculations, data needs to be placed in a histogram with a fixed number of bins. The number of bins can change the distribution of the histogram and thus the entropy. For data entropy, we chose six bins, since there were six distinct scalar values (one for each isosurface). For shading entropy and depth entropy, we chose the common default of 100. Finally, Ascent was also used to generate the isosurfaces and imagery for the user study.

5 Evaluation Approach

As discussed previously in Section 3.1, our method uses camera metrics to construct oracles that attempt to predict human behavior. Our evaluation for a given oracle measures the extent that oracle can successfully predict participant preferences in our data corpus. For each entry (D, C_i, C_j, R) in the corpus, our evaluation approach provides the oracle with the corresponding VQ metrics and records whether the oracle predicted the user would prefer C_i or C_j. If the oracle correctly predicts R, then it receives one point. If not, then it receives zero points. The oracle's score is the sum of these points over all 1096 entries, and the oracle with the highest score is the best, since it has made the highest number of correct predictions. Table 2 shows a notional example of this process.

Table 2: An example of evaluating an oracle on a notional corpus with four entries. Entries 1, 3, and 4 correspond to data set 1, while entry 2 corresponds to data set 2. This oracle would receive a total of two points, since it correctly predicts participant preferences for the first two entries, but is incorrect for the last two entries. Further, note that entries 1 and 4 involves the same combination of data set and cameras to compare, but the participants had different preferences, which is a situation that occurs in practice. This means that no oracle can achieve a perfect score for this corpus — the maximum score is three, since any oracle must make a poor prediction of user behavior for either entry 1 or entry 4.

Entry	Data Set	C_i	C_j	Participant Preference	Oracle Prediction
1	DS1	3	6	3	3
2	DS2	8	8	8	8
3	DS1	3	9	9	3
4	DS1	3	6	6	3

Despite having 1096 entries in our corpus, the maximum possible score is not 1096, because users sometimes disagree on which view is preferred (like entries 1 and 4 in Table 2). Therefore, none of our camera metric-based oracles should expect to get a score of 1096. We studied the corpus and determined the maximum achievable score is 952, i.e., when participants disagreed, the sum of the dissenting choices was 144.

Table 3: Correct predictions for each single-metric oracle. Percent correct is reported for both with respect to the maximum possible for our corpus (952 — see discussion in Section 5), and to the total number of entries (1096).

Metric	Correct Predictions	% (/952)	% (/1096)
Data Entropy	676	71.0%	61.6%
Shading Entropy	662	69.5%	60.4%
Maximum Depth	571	60.0%	52.1%
Depth Entropy	566	59.4%	51.6%
# of Visible Triangles	503	52.8%	45.9%
Visibility Ratio	502	52.7%	45.8%
Plemenos & Benayada	498	52.3%	45.4%
Viewpoint Entropy	492	51.6%	44.9%
VKL Distance	484	50.8%	44.2%
Projected Area	465	48.8%	42.4%

6 Results

This section evaluates how well camera metric-based oracles can predict user preference. It is organized into three sections:

- Section 6.1 evaluates single-metric oracles, i.e., metrics that make predictions using only one type of camera metric.
- Section 6.2 evaluates multi-metric oracles, i.e., metrics that make predictions using more than one type of camera metric.
- Section 6.3 considers the conditions where our top oracles can predict user behavior and where they cannot.

6.1 Single-Metric Oracles

The evaluation for each metric can be found in Table 3. The rate of correct prediction is surprisingly low for all metrics. An oracle that made random choices would be correct 50% of the time, and yet six of the metrics were unable to achieve this threshold. Certainly, these metrics (on their own) do not appear to be useful for the data sets in our corpus. The top performing metrics do include our new entropy-based metrics, although their rate of successful prediction is somewhat low. The best-performing metric, data entropy, is correct at slightly more than a 3-to-2 rate, although it does achieve 71% of the performance of a perfect oracle.

6.2 Multi-Metric Oracles

This multi-metric analysis begins by considering oracles that incorporate two metrics, with the top results listed in Table 4. This table demonstrates two important findings: (1) that our new entropy-based metrics are performing well as oracles and (2) that the method for combining metrics (i.e., **TIER**, **NORM**, or **NONE**) is not crucial. With respect to performance, each of the top oracles uses at least one of our entropy-based metrics. Further, the three combinations that involve two of our metrics (i.e., data entropy + shading entropy, data entropy + depth entropy, and shading entropy + depth entropy) rank as the top three for **NORM** and **NONE** and three of the top four for **TIER**. In fact, the top performer that does not have one of our entropy-based metrics are the 14$^{\text{th}}$ best performers for **NORM** and **TIER** (visible triangles+maximum depth for both cases) and the 13$^{\text{th}}$ best performer for **NONE** (VKL+maximum depth). With respect to combining metrics, while there is variation in the order and percentages, the overall trends are quite close: each of the top combinations are in the 63%-65% range and many of the same pairs of metrics are repeated across the table.

Table 5 continues the analysis with oracles that incorporate three metrics. This table shows highly similar results to the two-metric analysis: our entropy metrics are good performers and the method for combining metrics (**TIER**, **NORM**, **NONE**) is not all that significant. Once again, each of the top ten performers involves at

Table 4: This table displays results for two-metric oracles all three combination methods (**TIER**, **NORM**, **NONE**). There are 45 two-metric oracles, but only the best ten are shown for each combination method. The **TIER** method is prone to ties, and these ties were discarded, meaning the number of entries evaluated for **TIER** is lower than 1096. For example, the sum of the tiers for depth entropy and data entropy were equal for the camera pairs for 158 of the 1096 corpus entries, and so only the remaining 938 entries were considered. Further, its percentage is based on this lower number, i.e., $63.4\% \times 938$ means this oracle made the correct number of predictions 595 times. The number of entries considered for **TIER** evaluations ranged over the 45 combinations from 842 to 1016. For **NORM** and **NONE**, the percentages reflect all 1096 entries.

	NORM			TIER			NONE		
Rank	Metric 1	Metric 2	%	Metric 1	Metric 2	%	Metric 1	Metric 2	%
1	Data Ent.	Shading Ent.	64.4%	Data Ent.	Depth Ent.	63.4%	Data Ent.	Shading Ent.	65.2%
2	Data Ent.	Depth Ent.	64.1%	Data Ent.	Max Depth	63.4%	Shading Ent.	Depth Ent.	64.8%
3	Shading Ent.	Depth Ent.	62.2%	Data Ent.	Shading Ent.	62.8%	Data Ent.	Depth Ent.	63.0%
4	Data Ent.	Max Depth	62.0%	Shading Ent.	Depth Ent.	62.2%	Shading Ent.	PB	60.5%
5	Data Ent.	Visible △'s	59.4%	Data Ent.	Visible △'s	59.5%	Data Ent.	Vis. Ratio	60.1%
6	Shading Ent.	Visible △'s	57.0%	Data Ent.	Vis. Ratio	59.0%	Shading Ent.	Vis. Ratio	60.0%
7	Shading Ent.	Vis. Ratio	56.0%	Data Ent.	VKL	56.8%	Data Ent.	Max Depth	59.5%
8	Depth Ent.	Max Depth	56.7%	Shading Ent.	Max Depth	56.6%	Data Ent.	PB	59.3%
9	Data Ent.	Vis. Ratio	56.5%	Data Ent.	Viewpoint Ent.	56.3%	Shading Ent.	VKL	57.2%
10	Shading Ent.	Max Depth	55.7%	Depth Ent.	Max Depth	56.1%	Depth Ent.	Max Depth	55.9%

Table 5: This table displays results for three-metric oracles for all three combination methods (**TIER**, **NORM**, **NONE**). There are 120 three-metric oracles, but only the best ten are shown for each combination method. As discussed in Table 4's caption, the **TIER** method is prone to ties, and these ties were discarded. The number of entries considered for **TIER** evaluations ranged over the 120 combinations from 928 to 1051. Once again, the percentages for **NORM** and **NONE** reflect all 1096 entries. Finally, for formatting reasons, data entropy, depth entropy, and shading entropy are abbreviated DaE, DeE and ShE.

	NORM		TIER		NONE	
Rank	Metrics	%	Metrics	%	Metrics	%
1	DaE + ShE + DeE	65.7%	DaE + ShE + DeE	64.5%	DaE + ShE + DeE	68.0%
2	DaE + ShE + Max. Depth	64.1%	DaE + ShE + Vis. Ratio	63.3%	DaE + ShE + Vis. Ratio	65.0%
3	DaE + ShE + Vis. △'s	63.3%	DaE + ShE + Max. Depth	62.6%	DaE + ShE + PB	64.8%
4	DaE + ShE + Vis. Ratio	63.1%	DaE + ShE + VKL	61.8%	ShE + DeE + Vis. Ratio	64.5%
5	DaE + ShE + PB	61.4%	DaE + ShE + Visible △'s	61.7%	ShE + DeE + PB	63.8%
6	DaE + ShE + VKL	61.2%	DaE + DeE + Max. Depth	59.7%	DaE + ShE + VKL	62.9%
7	DaE + DeE + Max. Depth	60.7%	DaE + ShE + Viewpoint Entropy	59.6%	DaE + DeE + Vis. Ratio	61.8%
8	DaE + ShE + Projected Area	60.5%	ShE + DeE + Max. Depth	59.2%	DaE + DeE + PB	61.2%
9	DaE + ShE + Viewpoint Ent.	60.1%	DaE + DeE + Visible △'s	59.1%	ShE + DeE + Viewpoint Ent.	60.8%
10	ShE + DeE + Max. Depth	59.5%	DaE + DeE + Vis. Ratio	59.0%	ShE + Vis. Ratio + PB	60.5%

least one of our entropy-based metrics, and most involve two. Further, the top combination across all three combination methods is data entropy + shading entropy + depth entropy. One difference between the combination methods is that the **NONE** version yields the highest prediction rate (746 correct predictions with a maximum possible of 952). In terms of how the comparator (non-entropy) metrics performed, the top oracles were:

- **NORM** and **TIER** both had visible triangles + visibility ratio + maximum depth ranked as the 46$^{\text{th}}$ best combination (out of 120), with successful predictions 50.4% and 50.1% of the time, respectively.
- **NONE** had VKL + visibility ratio + maximum depth ranked as its 35$^{\text{th}}$ best combination (again out of 120), with successful predictions 54.3% of the time.

We repeated this analysis with four metrics, and found that 4-metric oracles were generally not as effective as the 3-metric oracles. One exception was the combination of data entropy, shading entropy, depth entropy, and PB (Plemenos & Benayada), which had a 68.4% winning percentage with **NONE**. That said, we are skeptical about the strength of this finding. On the one hand, PB does provide new insights that the entropy-based metrics do not have — it maximizes visible triangles and projected area per pixel, typically choosing viewpoints that "fill" the final image. On the other hand, the data in our corpus is noisy and the amount of improvement is small. Further, we are concerned that we run the risk of "reverse engineering" a result based on our corpus. In all, we conclude from this analysis that our entropy-based metrics do provide better prediction of user preference than previous methods, and also that the combination method is not important. We feel this conclusion is supported by the high rate that entropy-based metrics appear as top performers and in the invariance of the result across combination method.

6.3 Efficacy of Top Oracle

This section investigates how the top oracle (data entropy + shading entropy + depth entropy, combined with **NONE**) performed on the corpus.

Table 6 presents analysis about how the oracle performs for different types of cameras. Specifically, each camera is classified as "**POOR**," "**FAIR**," "**GOOD**," or "**VERY GOOD**" and the analysis considers how the oracle performs for each of the combinations (e.g., when the oracle is asked to choose between a "**POOR**" and "**GOOD**" camera). We classify the cameras based on their win percentage, i.e., the rate that an individual image was preferred by participants. For example, camera position #6 for the Mantle data set was one of the most preferred images, being preferred by users in 28 out of 31 comparisons, i.e., a "win percentage" of 90.3%. We label as follows: 0%-25% as **POOR**, 25%-50% as **FAIR**, 50%-75% as **GOOD**, and 75%-100% as **VERY GOOD**. While our oracle does not have access to either win percentage or these labels, they are useful for postmortem analysis of oracle behavior. In terms of findings, our oracle appears to be most effective at predicting user preference when **POOR** cameras are involved, with an 82% efficacy, which was the highest of any group by a large margin. When a participant is asked to choose between two cameras that are **GOOD** or **VERY GOOD**, our oracle is only 58% effective (175/300). Possibly these images

Table 6: Performance statistics for our top oracle. As an example of how to interpret this table, the data from the **GOOD**/**POOR** entry means that there were 222 instances in our corpus where our oracle was asked to choose between one camera that was **GOOD** and one camera that was **POOR**, and it correctly matched participant preference 188 times, which was an 85% success rate. The Sum column provides statistics about behavior for one camera grouping. For example, there were 470 instances in our corpus where at least one of the cameras was **POOR**, for which our oracle correctly matched participant preference 82% of the time. The Sum column involves double counting some of the corpus entries, e.g., entries with both **GOOD** and **FAIR** are counted in both the **GOOD** and **FAIR** sums. Some table cells have higher counts because the number of cameras for each type varies; there are 34 **GOOD** cameras and 22 of the other three types. Finally, table cells are colored by their performance: a success rate of 70% or more is colored green, 60%-70% is colored yellow, and less than 60% is colored pink.

	POOR	FAIR	GOOD	VERY GOOD	Sum
POOR	82% (23/28)	75% (55/73)	85% (188/222)	80% (118/147)	82% (384/470)
FAIR	75% (55/73)	60% (30/50)	54% (88/162)	61% (69/114)	61% (242/399)
GOOD	85% (188/222)	54% (88/162)	58% (57/99)	60% (107/177)	67% (440/660)
VERY GOOD	80% (118/147)	61% (69/114)	60% (107/177)	46% (11/24)	66% (306/462)

are both adequate to the participant, and so other factors, such as esthetics, become more important. Overall, this table shows that we perform relatively similarly for all types of cameras. Moreover, it does not show evidence that our oracle is under-performing for certain types of comparisons.

Table 7: The rate of successful predictions by our top oracle per data set.

Data Set	Prediction Rate
Asteroid	74.2%
Constit	52.2%
ExaAM	76.5%
Fluid Dynamics	80.8%
Hurricane	60.3%
Mantle	79.1%
Miranda	52.5%
S3D-N2	57.7%
S3D-UVEL	72.7%
ExaSky Nyx	71.5%

Table 7 presents prediction rate by data set. Three of the data sets — Constit, Miranda, and S3D-N2 — have success rates below 60%, and a fourth, Hurricane, is just above 60%. The remainder of the data sets are at or above the oracle average. Visual inspection of the images for these four data sets shows:

- Constit has many views that are similar. Participants preferred those with empty space between the isosurfaces, presumably the other views created confusion with occlusion issues. None of the entropy-based metrics would assist with this issue.
- Hurricane is composed of several layers with high rates of occlusion, undoubtedly causing issues for both data entropy and depth entropy. Additionally, this data set is easily identifiable and an uncommon choice of esthetics may influence user preference.
- Miranda has blue/orange surfaces at the boundary of the volume that are (in the opinion of the authors) not as interesting as the isosurfaces at the mixing layer. That said, data entropy rewards showing more of these blue/orange surfaces. When we modified our oracle to ignore data entropy, the prediction rate went to 69.7%, in line with the oracle average.
- S3D-N2 has small features that are clearly visible from some images, but not from others. Users preferred the images with these features, but the data entropy calculation did not reflect their presence since its calculations were dominated by the other surfaces.

All of these observations suggest possible future improvements for an oracle-based scheme.

7 Conclusion

The research premise of this study is that the entropy-based metrics are good predictors, especially in combination, and we feel our results provide strong evidence to support this premise. We do believe that we could look further at ways to combine metrics (such as weighted combinations) and decision-tree type oracles ("if the data entropy is better then choose this camera, else look at shading entropy...") and optimize those approaches for this given corpus. But such optimizations would have to be verified on a different corpus with different data set and users; we view this as future work. We also believe the shortcomings identified with Constit, Hurricane, Miranda, and S3D-N2 suggest future avenues of improvement.

We (surprisingly) recommend the **NONE** combination method of adding the three scores together. While such an approach creates apples-to-oranges concerns when involving non-entropy metrics, it is appropriate when adding three entropy metrics together. Our motivation for this recommendation is not because **NONE** got the highest prediction percentage, but because it requires no additional knowledge. **NORM** and **TIER** require evaluating multiple cameras to establish a baseline for what should go in a "good" or "bad" tier or what should be normalized to "1" or "10." **NONE** does not require these extra calculations, which is an advantage in an in situ setting.

There are two major areas of future work: (1) how the study itself can be improved and extended, and (2) how the results of this study can be used for automating in situ camera placement. With respect to improving the study, we would like to continue to expand our corpus of data and evaluate additional metrics and oracles. In particular, our corpus consisted of multi-level isosurface imagery, and other visualizations may have differing results. Further, we limited our consideration of combined metrics due to concerns of overfitting, but an additional corpus would enable us to optimize combinations on one corpus and validate on the other. Other extensions could include enhancing the survey, for example to include stereo imagery, or enhancing the overall approach, for example considering variation in isolevels as well as camera position. With respect to in situ automation, future work includes parallelizing the viewpoint quality metrics used in this paper for a distributed-memory setting. Further, there are a number of research avenues that need exploring, including evaluating performance at scale; devising a search algorithm to quickly find a data set's global optimal viewpoint; and exploring the time-varying aspects of the best viewpoint — is it consistent or erratic?, and if it is erratic, then should camera movements be discouraged to make a more coherent animation?

Acknowledgments

This research was supported by the Exascale Computing Project (17-SC-20-SC), a collaborative effort of the U.S. Department of Energy Office of Science and the National Nuclear Security Administration.

REFERENCES

[1] Hurricane isabel simulation data. p. online, 2004.

[2] A. S. Almgren, J. B. Bell, M. J. Lijewski, Z. Luki, and E. Van Andel. Nyx: A massively parallel amr code for computational cosmology. *The Astrophysical Journal*, 765(1):39, Feb 2013.

[3] J. Belak, J. Turner, and E. T. Team. Exaam: Additive manufacturing process modeling at the fidelity of the microstructure. In *APS March Meeting Abstracts*, vol. 2019, pp. C22–010, 2019.

[4] R. E. Blahut. *Principles and Practice of Information Theory*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1987.

[5] V. Blanz, M. J. Tarr, and H. H. Bülthoff. What object attributes determine canonical views? *Perception*, 28 5:575–99, 1999.

[6] X. Bonaventura, M. Feixas, and M. Sbert. Viewpoint information. *21st International Conference on Computer Graphics and Vision, GraphiCon'2011 - Conference Proceedings*, 01 2011.

[7] X. Bonaventura, M. Feixas, M. Sbert, L. Chuang, and C. Wallraven. A survey of viewpoint selection methods for polygonal models. *Entropy*, 20(5), 2018.

[8] U. Bordoloi and H. Shen. View selection for volume rendering. In *16th IEEE Visualization Conference, VIS 2005, Minneapolis, MN, USA, October 23-28, 2005*, pp. 487–494, 2005.

[9] J. Burbea and C. Rao. On the convexity of some divergence measures based on entropy functions. *IEEE Transactions on Information Theory*, 28(3):489–495, May 1982.

[10] R. A. Carson, S. R. Wopschall, and J. A. Bramwell. ExaConstit, Aug 2019.

[11] A. W. Cook, W. H. Cabot, P. L. Williams, B. J. Miller, B. R. de Supinski, R. K. Yates, and M. L. Welcome. Tera-scalable algorithms for variable-density elliptic hydrodynamics with spectral accuracy. In *SC'05: Proceedings of the 2005 ACM/IEEE Conference on Supercomputing*, pp. 60–60. IEEE, 2005.

[12] C. D. Correa and K.-L. Ma. Visibility-driven transfer functions. In *2009 IEEE Pacific Visualization Symposium*, pp. 177–184, 2009.

[13] T. M. Cover and J. A. Thomas. *Elements of Information Theory (Wiley Series in Telecommunications and Signal Processing)*. Wiley-Interscience, New York, NY, USA, 2006.

[14] H. Dutaac, C. Cheung, and A. Godil. A benchmark for best view selection of 3d objects. *3DOR'10 - Proceedings of the 2010 ACM Workshop on 3D Object Retrieval, Co-located with ACM Multimedia 2010*, pp. 45–50, 01 2010.

[15] M. Feixas, M. Sbert, and F. González. A unified information-theoretic framework for viewpoint selection and mesh saliency. *ACM Trans. Appl. Percept.*, 6(1):1:1–1:23, Feb. 2009.

[16] S. Gumhold. Maximum entropy light source placement. In *ACM SIGGRAPH 2002 Conference Abstracts and Applications*, SIGGRAPH '02, p. 215. Association for Computing Machinery, New York, NY, USA, 2002.

[17] L. Itti, C. Koch, and E. Niebur. A model of saliency-based visual attention for rapid scene analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(11):1254–1259, Nov 1998.

[18] G. Jenks. Optimal data classification for choropleth maps occasional paper no 2. *University of Kansas, Department of Geography*, 1977.

[19] T. Kamada and S. Kawai. A simple method for computing general position in displaying three-dimensional objects. *Computer Vision, Graphics, and Image Processing*, 41(1):43 – 56, 1988.

[20] J. Kuhnert. Meshfree numerical schemes for time dependent problems in fluid and continuum mechanics. *Advances in PDE modeling and computation*, pp. 119–136, 2014.

[21] M. Larsen, J. Ahrens, U. Ayachit, E. Brugger, H. Childs, B. Geveci, and C. Harrison. The alpine in situ infrastructure: Ascending from the ashes of strawman. In *Proceedings of the In Situ Infrastructures on Enabling Extreme-Scale Analysis and Visualization*, ISAV'17, pp. 42–46. ACM, New York, NY, USA, 2017.

[22] C. H. Lee, A. Varshney, and D. W. Jacobs. Mesh saliency. In *ACM SIGGRAPH 2005 Papers*, SIGGRAPH '05, pp. 659–666. ACM, New York, NY, USA, 2005.

[23] T.-Y. Lee, O. Mishchenko, H.-W. Shen, and R. Crawfis. View point evaluation and streamline filtering for flow visualization. pp. 83–90, 03 2011.

[24] J. Lin. Divergence measures based on the shannon entropy. *IEEE Transactions on Information Theory*, 37(1):145–151, Jan 1991.

[25] R. Naraoka, I. Fujishiro, Y. Takeshima, and S. Takahashi. Locating an optimal light source for volume rendering. *Journal of the Visualization Society of Japan*, 27:45–46, 07 2007.

[26] D. Page, A. Koschan, S. R. Sukumar, B. Abidi, and M. Abidi. Shape analysis algorithm based on information theory. vol. 1, pp. 229–232, 01 2003.

[27] S. E. Palmer, E. Rosch, and P. Chase. Canonical perspective and the perception of objects. 1981.

[28] J. Patchett and G. Gisler. Deep water impact ensemble data set. 4, 2017.

[29] D. Plemenos. *Contribution à l'étude et au développement des techniques de modélisation, génération et visualisation de scènes : le projet multiformes*. PhD thesis, 1991. 1991NANT2041.

[30] D. Plemenos and M. Benayada. Intelligent display in scene modelling. new techniques to automatically compute good views. In *GraphiCon'96*. Saint Petersburg (Russia), July 1996.

[31] O. Polonsky, G. Patané, S. Biasotti, C. Gotsman, and M. Spagnuolo. What's in an image? *The Visual Computer*, 21(8):840–847, Sep 2005.

[32] M. Sbert, M. Feixas, J. Rigau, I. Viola, and M. Chover. Applications of information theory to computer graphics. In *Eurographics*, 2002.

[33] M. Sbert, D. Plemenos, M. Feixas, and F. González. Viewpoint quality: Measures and applications. In *Proceedings of the First Eurographics Conference on Computational Aesthetics in Graphics, Visualization and Imaging*, Computational Aesthetics'05, pp. 185–192. Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, 2005.

[34] A. Secord, J. Lu, A. Finkelstein, M. Singh, and A. Nealen. Perceptual models of viewpoint preference. *ACM Trans. Graph.*, 30(5):109:1–109:12, Oct. 2011.

[35] M. H. Shahnas, W. R. Peltier, Z. Wu, and R. Wentzcovitch. The high-pressure electronic spin transition in iron: Potential impacts upon mantle mixing. *Journal of Geophysical Research: Solid Earth*, 116(B8), 2011.

[36] D. Sokolov and D. Plemenos. Viewpoint quality and scene understanding. In *Proceedings of the 6th International Conference on Virtual Reality, Archaeology and Intelligent Cultural Heritage*, VAST'05, pp. 67–73. Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, 2005.

[37] S. L. Stoev and W. Strasser. A case study on automatic camera placement and motion for visualizing historical data. In *IEEE Visualization, 2002. VIS 2002.*, pp. 545–548, Oct 2002.

[38] S. Takahashi, I. Fujishiro, and Y. Takeshima. Interval volume decomposer: a topological approach to volume traversal. In R. F. Erbacher, J. C. Roberts, M. T. Grohn, and K. Borner, eds., *Visualization and Data Analysis 2005*, vol. 5669, pp. 103 – 114. International Society for Optics and Photonics, SPIE, 2005.

[39] S. Takahashi, I. Fujishiro, Y. Takeshima, and T. Nishita. A feature-driven approach to locating optimal viewpoints for volume visualization. In *VIS 05. IEEE Visualization, 2005.*, pp. 495–502, Oct 2005.

[40] M. J. Tarr and D. J. Kriegman. What defines a view? *Vision Research*, 41(15):1981 – 2004, 2001.

[41] S. Treichler, M. Bauer, A. Bhagatwala, G. Borghesi, R. Sankaran, H. Kolla, P. Mccormick, E. Slaughter, W. Lee, A. Aiken, and J. H. Chen. S3d-legion: An exascale software for direct numerical simulation of turbulent combustion with complex multicomponent chemistry. 11 2017.

[42] P.-P. Vázquez, M. Feixas, M. Sbert, and W. Heidrich. Viewpoint selection using viewpoint entropy. In *Proceedings of the Vision Modeling and Visualization Conference 2001*, VMV '01, pp. 273–280. Aka GmbH, 2001.

[43] T. Vieira, A. L. Bordignon, A. Peixoto, G. Tavares, H. Lopes, L. Velho, and T. Lewiner. Learning good views through intelligent galleries. *Comput. Graph. Forum*, 28:717–726, 2009.

[44] I. Viola, M. Feixas, M. Sbert, and M. E. Groller. Importance-driven focus of attention. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):933–940, 2006.

[45] I. Viola, A. Kanitsar, and M. Groller. Importance-driven volume rendering. In *IEEE Visualization 2004*, pp. 139–145, 2004.

[46] Y. Xiaoling and W. Yanni. A view selection method based on particle

swarm optimization. In *2015 International Conference on Computers, Communications, and Systems (ICCCS)*, pp. 69–72, 2015.