Importance-Driven Focus of Attention

Ivan Viola, *Member, IEEE*, Miquel Feixas, *Member, IEEE*,
Mateu Sbert, *Member, IEEE*, and Meister Eduard Gröller, *Member, IEEE*

Abstract—This paper introduces a concept for automatic focusing on features within a volumetric data set. The user selects a focus, i.e., object of interest, from a set of pre-defined features. Our system automatically determines the most expressive view on this feature. A characteristic viewpoint is estimated by a novel information-theoretic framework which is based on the mutual information measure. Viewpoints change smoothly by switching the focus from one feature to another one. This mechanism is controlled by changes in the importance distribution among features in the volume. The highest importance is assigned to the feature in focus. Apart from viewpoint selection, the focusing mechanism also steers visual emphasis by assigning a visually more prominent representation. To allow a clear view on features that are normally occluded by other parts of the volume, the focusing for example incorporates cut-away views.

Index Terms—Illustrative visualization, volume visualization, interacting with volumetric datasets, characteristic viewpoint estimation, focus+context techniques

✦

1 INTRODUCTION

Visual presentation of underlying non-graphical data is one of the most important visualization aims. It serves as a communication medium and can be motivated by, e.g., educational, infographics, or commercial purposes. This paper helps in improving visual presentation of structures within complex three-dimensional data such as computed tomography (CT) data.

CT scanning allows insight into different species, materials, or bodies. One of the most important application areas is medical diagnostic imaging. Software for medical workstations in general include the broadest spectrum of functionality for handling volumetric CT data sets. This includes visualization, image processing, measurements, or (semi-)automatic diagnosis estimation. Medical workstations, however, are designed mostly for visual analysis in diagnostic scenarios, rather than for presentation purposes. Increasingly the topic of visual presentations is becoming important in the communication between medical experts or between the medical staff and the patient. Therefore functionality for presentation purposes will become more important for medical workstations in the future.

Current volume visualization systems require a lot of expertise from the user. For example many widgets to design a suitable transfer function (mapping tissue density to color and opacity values) are not intuitive for the inexperienced user. Our work is motivated by the fact that currently none of the commercially or publicly available visualization systems allows the user high-level interactions such as "Show me this interesting part of the volumetric data set and then show me the next interesting part." Our framework allows an automatic focus of attention on interesting objects. The user's only required (not limited to) interaction is to select an object of interest from a set of pre-segmented objects. The framework smoothly navigates the view to clearly see the characteristics of the focus object. Additionally, the focus object is visually emphasized for easy discrimination from the context. Example images that illustrate focus of attention for insight into a human hand data and a human torso data are shown in Figures 1 and 2.

- *I. Viola and M. E. Gröller are with Institute of Computer Graphics and Algorithms, Vienna University of Technology, Austria. Viola is at the time of publication affiliated with University of Bergen, Norway.*
  E-mail: {viola | meister}@cg.tuwien.ac.at.
- *M. Feixas and M. Sbert are with Institute of Informatics and Applications, University of Girona, Spain.*
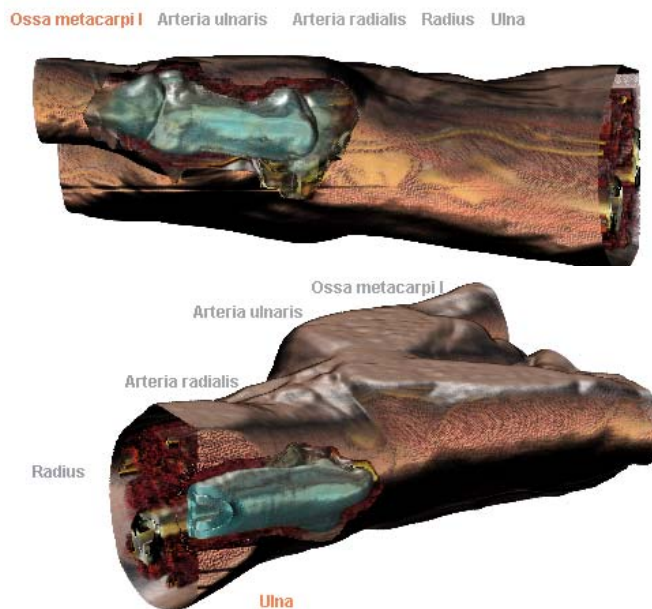 *E-mail: {feixas | mateu}@ima.udg.es.*

Fig. 1. Visual inspection of different parts within a human hand data.

The main contribution of the paper is a concept of focus of attention for interactive volume visualization. A characteristic viewpoint is selected in combination with a visually pleasing discrimination of the focus from the context information. By changing the object of interest, both viewpoint settings and visual parameters are smoothly modified to put emphasis on the newly selected object of interest. The second contribution is the introduction of an information-theoretic framework for characteristic viewpoint estimation in volumetric data sets with pre-segmented objects. Both frameworks, i.e., the interactive focusing approach and the identification of characteristic viewpoints, are controlled by an intuitive importance distribution among structures within the volumetric data.

The paper is organized as follows: Section 2 describes previous work related to importance-driven focus of attention. Section 3 describes the framework for focusing. Technical details of obtaining characteristic viewpoints are discussed in Section 4. Interaction aspects of focusing are presented in Section 5. Implementation issues and performance are discussed in Section 6. We draw conclusions and
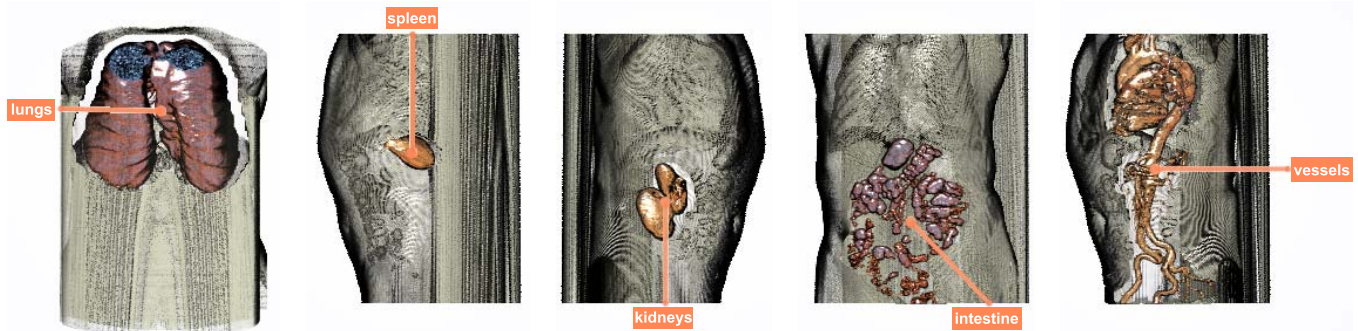
Fig. 2. Focus of attention applied to the visual inspection of organs inside the human torso.

summarize the paper in Section 7. Finally in Section 8 some future work is shortly mentioned.

2 RELATED WORK

Focus of attention has often been used in visualization to catch the user's attention. It has many different occurrences. We will first review relevant previous work in the area of focus+context visualization and user interaction. The second part of this section reviews recent work on optimal viewpoint estimation.

The depth of field effect is a focus of attention technique from photography that inspired Kosara et al. [6] to propose a semantic depth of field (SDOF). In their work they have shown that the degree of sharpness determines the speed of drawing human attention in otherwise blurry environments. They have applied their technique in various fields of information visualization.

A focus+context method for displaying volumetric data has been proposed in our previous work on importance-driven volume visualization [15]. The importance classification has been introduced for specifying view-dependent visual representations to reveal occluded structures. This is in the spirit of cut-away views and ghosted views known from traditional illustrations.

Several focus+context techniques have been included into VolumeShop, a publicly available interactive volume visualization system from our group [2]. The functionality of VolumeShop is intended to provide a tool for presenting and communicating the data being visualized. Similarly to the *fan* feature in VolumeShop, Tearum et al. [10] present contextual super-resolution close-ups from illustration applied to medical volume visualization.

Another publicly available visualization system including functionality for visual presentation and communication has been proposed by Svahkine et al. [9]. An interesting aspect incorporated in their system is the level of expertise of the user. This has two implications for the design of the system. First, the user interface and the widgets are customized according to the user-expertise. A non-expert user has a very simple user interface allowing limited flexibility, whereas an expert has much higher flexibility with advanced tools such as a transfer function editor. Second, the level of user expertise implies also different visualization results. An easy to understand visualization is targeted to a non-expert user and a more *direct* visualization is targeted to the expert.

Design of camera path is a part of visual presentation. A recent technique on camera motion for the design review of polygonal models has been proposed by Burtnyk et al. [3]. The aim of their ShowMotion system is visually pleasing camera paths inspired by cinematographic effects. Camera paths are reduced to high-quality motions only, leaving out transitions from one feature to another which are replaced by fade-out and fade-in operations.

Good viewpoint selection is crucial for an effective focus of attention. Viewpoint selection has already been applied to several domains in computer graphics, examples include image-based modeling [14],

volume visualization [1, 11], or mesh saliency [7]. Different measures for viewpoint evaluation have been used in these fields.

Vázquez et al. [13] have introduced the *viewpoint entropy* (VE) as a measure for viewpoint quality evaluation. This measure has been designed primarily for polygonal data, where the best viewpoint is defined as the one that has maximum entropy. Taking into account the background information, this technique may be used for indoor and outdoor scenes as well. VE, based on the Shannon entropy [4], has been defined as

$$H_v = -\sum_{i=0}^{N_p} \frac{a_i}{a_t} \log \frac{a_i}{a_t},\qquad(1)$$

where N_p is the number of polygons of the scene, a_i is the projected area of polygon i over the sphere of directions centered at viewpoint v, a_0 represents the projected area of the background, and $a_t = \sum_{i=0}^{N_p} a_i$ is the total area of the sphere. The maximum entropy is obtained when a certain viewpoint can see all the polygons with the same projected area a_i. VE for polygonal data has been recently extended to volumetric scalar data [1], by substituting the area visibility distribution by the the distribution obtained from the quotient between the voxel visibility and the voxel importance (*noteworthiness* factor). This work has additionally suggested information-theoretic measures for clustering views according to similarity using the Jensen-Shannon divergence. They also suggested an optimal viewpoint estimation scheme for time-varying data.

It has been shown recently by Sbert et al. [8] that VE is very sensitive to triangulation. Thus, the maximum entropy is achieved by the viewpoint that *sees* areas with a very fine triangulation. They proposed a new viewpoint-quality measure for polygonal data based on the *Kullback-Leibler (KL) distance* [4] (Equation 6). The *viewpoint KL distance* KL_v has been defined as the distance between the normalized distribution of projected areas and the normalized distribution of the actual areas:

$$KL_v = \sum_{i=1}^{N_p} \frac{a_i}{a_t} \log \frac{\frac{a_i}{a_t}}{\frac{A_i}{A_T}},\qquad(2)$$

where A_i is the actual area of polygon i and $A_T = \sum_{i=1}^{N_p} A_i$ is the total area of the scene or object. In this case, the background is not taken into account. The minimum value zero would be obtained when the normalized distribution of the projected areas is equal to the normalized distribution of the actual areas. In this framework, the best-quality views correspond to views with minimal KL distance. One drawback of this measure is that many non-visible or poorly visible polygons in a model can distort the quality of the measure.

3 IMPORTANCE-DRIVEN FOCUSING

Before going into technical details of our work we would like to *focus* the reader's *attention* on several considerations we have made during designing our framework. To get a clear high-level overview on the

framework functionality, we briefly present the processing framework. Technical details follow in Sections 4 and 5.

3.1 Framework Design Considerations

In our framework we consider two different aspects of focusing:

Focus discrimination: Focus of attention is a visual discrimination of interesting objects from other elements in the scene. It is realized through a visual emphasis of the object of interest while other objects presented as context are suppressed. In general a discrimination of the focus from the context can be achieved by different *levels of sparseness* in their visual representation [15]. The focus is represented very *densely* while the context gets a more *sparse* visual representation. Levels of sparseness can be designed in many ways. In photography for example, a very effective technique for object discrimination is the sharpness of the object of interest. Very sharp objects are automatically perceived as being in focus, more blurry objects are contextual information. Levels of sparseness are in this case different sharpness levels. In our framework we use opacity, color brightness, and saturation to discriminate the most interesting objects from the rest.

Characteristic view: In addition to visual discrimination, objects in focus have to be shown from a good and characteristic view where most of the focus structures are perceivable. The most interesting object must not be occluded by less relevant parts. If possible the focus should be in front of other features. In case that the feature of interest is always occluded by other features, cut-away views or other concepts from illustration should be included into the visualization. In this case it is important that the cut-away region does not entirely remove other interesting objects. If possible, only the least relevant objects are cut away. Furthermore a proper orientation of the up-vector of the viewpoint and a proper positioning of the focus are important to consider in the viewpoint specification. This helps to adhere to aesthetical criteria of composition such as the rule of thirds [5]. All mentioned aspects indicate that a proper viewpoint setting is important for the focus of attention.

3.2 Focus of Attention Framework

Our previous work [15] used an explicit importance classification for focus+context visualization inspired by techniques known from traditional illustration. In the following we give an overview how to use importance classification concept for automatic focusing at objects of interest. The entire framework is shown in Figure 3. The central element is the **importance distribution** of objects. It serves as a controlling mechanism for both parts, i.e., for the **characteristic viewpoint estimation** which is done in a pre-processing step as well as **interactive focus of attention**.

Finding a viewpoint where the characteristics of a specific feature are clearly visible naturally requires the **visibility estimation** of the feature under specific viewing settings. In our case, i.e., for objects within the volumetric data set, this process is rather time-consuming as it requires ray casting of the whole data set from various viewpoints. Computing the visibility of features on-the-fly during interaction will strongly limit interaction possibilities. The visibility of features depends on their visual representation. For applications where a frequent change of visual representations is not relevant, the visibility estimation can be easily treated as a pre-processing step, which is executed only once.

In our focus of attention framework we compute the visibility of an object as its contribution on the finally rendered image. This computation is based on the opacity contribution of each voxel belonging to the object. Additionally two weights influence the visibility of an object, i.e., **image-space weight** and **object-space weight**. Image-space weight penalizes the visibility of objects when they are located outside the image center. Object-space weight assigns higher visibility to objects which are close to the viewing plane and penalizes those that are more far away.

Object visibility is then mapped to a *conditional probability* of the object for a given viewpoint. These values are used for computation of good viewpoints for a given object. We use for this a novel
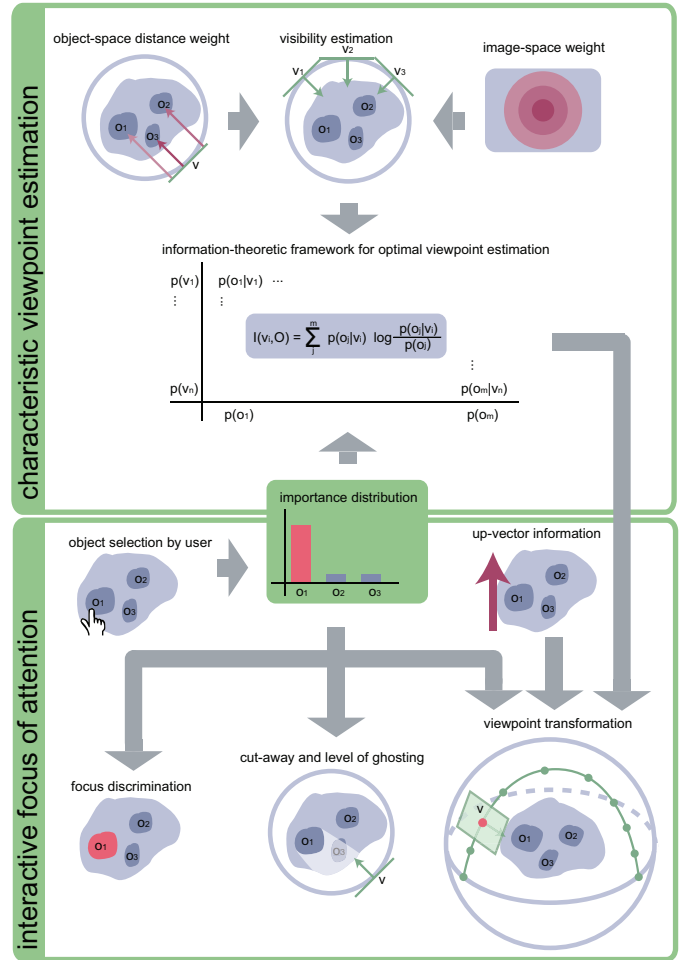

Fig. 3. Focus of attention framework: First the characteristic viewpoints for each feature are identified in a pre-processing step. During the user interaction viewpoint settings and visual representations are changed to bring the object of interest into focus. Both stages of the framework are controlled by the importance distribution of objects.

information-theoretic framework for optimal viewpoint estimation combined with object importance information as described in detail in Section 4.

With selecting visual representations of segmented objects and by identifying representative viewpoints, the crucial information to perform interactive focus of attention is available. The importance distribution is in the interactive part a direct mapping of user's interest. Importance is directly mapped to **focus discrimination** and **level of ghosting**. The **viewpoint transformation** is also controlled by importance distribution smoothly changing to characteristic viewpoints obtained in the pre-processing step. To preserve natural orientations of the viewpoints the framework includes the information about the **up-vector** of the volume (e.g., in the case of the human anatomy, up-vector is pointing upwards relative to the feet). A more detailed discussion on the interactive focus of attention is given later in Section 5.

4 CHARACTERISTIC VIEWPOINT ESTIMATION

In this section, we describe our approach for selecting a characteristic viewpoint for a particular object. This information is then used in the interactive focus of attention. First, we determine the visibility of structures within the volumetric data considering their visual representations. Then we use the visibility as input to the new information-theoretic framework. This framework integrates per-object importance classification, which allows to estimate characteristic viewpoints for an object within the volume.

4.1 Visibility Estimation

The first step for a viewpoint evaluation is the estimation of per-object visibility. We use a simple scheme for visibility evaluation, taking into account the opacity contribution of voxels on the rendered image. The evaluation of the visibility is done in a ray-casting step. For each sample i along a ray r we evaluate its visibility $v(r,i) = v(r,i-1)\alpha(r,i)$, where $\alpha(r,i)$ is the resampled opacity value at the given sample position i. We implement nearest neighbor and linear interpolation resampling schemes. The visibility of a voxel is given as the sum of visibilities of all resampled points the voxel is contributing to in the resampling step. In case of nearest neighbor interpolation we simply sum the ray sample visibilities *belonging* to this voxel. In case of linear interpolation, we perform a linear *distribution* of the ray sample visibility among all eight surrounding voxels. The quality difference of the visibility estimation between nearest neighbor and linear interpolation, however, is rather low. The performance difference on the other hand is significant, so the nearest neighbor resampling is preferred in our framework.

The sum of voxel visibilities belonging to a particular feature, estimates the visibility of this feature. We are using non-binary object classifications and a particular voxel may contribute to a number of different features simultaneously. The voxel visibility is simply multiplied by a factor that defines how much the voxel contributes to a particular object.

In our focus of attention framework, we also change the visual representation of the object of interest. This means that the visual representation is not constant during the time of interaction. This has to be taken into account while computing visibilities. Therefore we compute the visibility for each *active* object, i.e., object in focus. This means, for each viewpoint we get $(n+1)$ different visibility values for n objects. Each object is set once as active object plus once the visibility is computed with no selected active object. When we search for the characteristic viewpoint of a particular object, we use those visibilities where this object has been the active object.

One problem that arises when computing the visibility of objects, is that some features may be completely occluded by other features. This is caused by very dense visual settings. This means that there is no viewpoint from which the feature is clearly visible, or all viewpoints are equally good or bad. In order to deal with this problem, we have optionally included cut-away views in the visibility estimation. Here the active object is visible from all viewpoints as the volume region in front of this object is not visible at all.

The above described visibility evaluation does not consider the location of features in image and object space. To draw attention to a feature, it is important that it is located close to the center of the image. Therefore we give more prominence to rays in the center of the image. Each ray's contribution to the visibility of objects and background is scaled by an *image-space weight*. This weight is largest in the center of the image and is decreasing with the distance from the center.

Another weight that contributes to the visibility estimation takes the distance of an object to the viewer into account. This *object-space weight* is a high value when the feature is close to the viewing plane and low when it is farther away. This weight is especially important when cut-away views in the visibility estimation are enabled. The motivation is to penalize views that cut away larger parts of volume to *see* the active object.

The overall concept of characteristic viewpoint estimation driven by an importance distribution is illustrated in the upper stage in Figure 3. The importance distribution and the visibility of each object for the given visual representations are input parameters of the information-theoretic framework. This framework will be described in detail in the next section.

4.2 Information-Theoretic Framework for Optimal Viewpoint Estimation

After the visibility of each object under different visual settings and viewpoints has been computed, the characteristic viewpoint estimation can be performed. Our viewpoint selection approach is using the mutual information of the *information channel* defined in terms of *visibility* between a set of viewpoints and the objects of a volumetric data set. This new measure shows a better behavior and robustness than the previous viewpoint entropy [13]. An information channel between two random variables (input and output) is characterized by a probability transition matrix which determines the output distribution given the input. For more information on information-theoretic measures, please refer to Cover and Thomas [4].

Our framework works well for volumetric objects as segmented volume regions. Taking a voxel as a basic object element would lead to very high memory consumption. This will also be the case using previously suggested viewpoint quality measures. Therefore our basic element is an object (a feature) instead of a voxel.

The framework naturally integrates per-object importance classification. By changing the importance distribution among objects, the results of viewpoint evaluation also change to have a characteristic view on the feature of highest importance. Setting the importance to be equal for all objects, characteristic views for the entire volume are achieved.

We formalize our viewpoint selection framework defining a channel $V \rightarrow O$ between two random variables, which represent, respectively, a set of viewpoints V and a set of objects O of a volumetric data set. Viewpoints will be indexed by v and objects by o. The marginal probability distribution of V is given by $p(v) = \frac{1}{N_v}$, where N_v is the number of viewpoints, i.e., we assign the same probability to each viewpoint. The conditional (or transition) probabilities $p(o|v)$ are given by the normalized visibility of each object from each viewpoint, i.e., $\sum_o p(o|v) = 1$. From these data, the marginal probability distribution of O is given by

$$p(o) = \sum_v p(v)p(o|v) = \frac{1}{N_v}\sum_v p(o|v), \qquad (3)$$

which is the average visibility of each object obtained from the set of viewpoints.

The *mutual information* [4] between V and O expresses the degree of *dependence* or *correlation* between the set of viewpoints and the data set, and is defined as

$$I(V,O) = \sum_v p(v) \sum_o p(o|v) \log \frac{p(o|v)}{p(o)} = \frac{1}{N_v}\sum_v I(v,O), \qquad (4)$$

where we define

$$I(v,O) = \sum_o p(o|v) \log \frac{p(o|v)}{p(o)} \qquad (5)$$

as the *viewpoint mutual information* (VMI). VMI represents the degree of dependence between viewpoint v and the set of objects. In our framework, the quality of a viewpoint is given by $I(v,O)$ and the best viewpoint is defined as the one that has *minimum mutual information*. High values of the measure mean a high dependence between viewpoint v and the dataset, indicating a highly *coupled* or *dependent* view between, for instance, the viewpoint and a small number of objects with low average visibility. On the other hand, low values correspond to more *representative* or *independent* views, showing the maximum possible number of objects in a balanced way. Thus, we will select the viewpoints that minimally contribute to the mutual information of the channel.

It is worth observing that VMI (Equation 5) can be expressed as a *Kullback-Leibler distance* [4]. The Kullback-Leibler (KL) distance between two probability distributions p and q defined over the same set is given by

$$KL(p|q) = \sum_x p(x) \log \frac{p(x)}{q(x)}, \qquad (6)$$

and is a divergence measure between the *true* probability distribution p and the *target* probability distribution q. Thus, in Equation 5, $p(o)$ is the *target* distribution and plays the role of the *optimal* distribution as we want that $p(o|v)$ comes close to $p(o)$ to obtain the best views. On

the other hand, this role agrees with intuition since $p(o)$ is the average visibility of object o over all viewpoints, i.e., the *mixed distribution* of all views, and we can think of $p(o)$ as representing, with a single distribution, the knowledge about the scene. If we look for a good set of views within the set of viewpoints (in particular for the single best one), we will obtain the most representative set by selecting the views such that their mixing minimizes the distance to the target distribution. That is, this mixing will provide us with a *balanced* view of the dataset.

In addition, one important advantage of VMI over VE is its robustness to deal with any type of discretization or resolution of the dataset. It has been shown for polygonal data that the VE value is very sensitive to the discretization of the objects [8]. Analogously, a volumetric object with an extremely refined mesh will attract the attention of the measure. On the other hand, VMI will be near insensitive to changes in the voxel resolution. The behavior of both measures with respect to discretization can be deduced from the mathematical analysis of VE and VMI. For instance let's assume that a regular object o of the dataset is subdivided into two equal parts o_1 and o_2 such that $p(o_1|v) = p(o_2|v)$, $p(o_1) = p(o_2)$, $p(o|v) = p(o_1|v) + p(o_2|v)$ and $p(o) = p(o_1) + p(o_2)$. Assuming that only the term referred to object o can change in the formulas for VE and VMI, we analyze its variation after subdivision. Thus, for VE, given by $H_v = -\sum_o p(o|v) \log p(o|v)$ using our notation, we obtain that

$$
\begin{aligned}
-p(o|v) \log p(o|v) \quad &< \quad -p(o_1|v) \log p(o_1|v) - p(o_2|v) \log p(o_2|v) \\
&= \quad p(o|v) - p(o|v) \log p(o|v).
\end{aligned}
$$

Therefore, VE increases with a value $p(o|v)$ after the subdivision. On the other hand, for VMI,

$$
p(o|v) \log \frac{p(o|v)}{p(o)} = p(o_1|v) \log \frac{p(o_1|v)}{p(o_1)} + p(o_2|v) \log \frac{p(o_2|v)}{p(o_2)}.
$$

Thus, VMI remains invariant to this subdivision. Hence, if we compare both measures for a setting with different discretizations, mutual information will give similar results and VE will show an erratic behavior. For example, if we consider a fly eye either as a grouping of objects (facets) or simply a unique object, viewpoint entropy can select completely different views, while VMI will be almost insensitive to the change of model.

4.3 Incorporating the Importance Distribution

Due to the fact that VMI represents the distance between the projected visibility distribution $p(o|v)$ from viewpoint v and the target distribution $p(o)$, VMI can be easily generalized to incorporate importance.

Adding importance to our scheme means simply modifying the target function. The ideal viewpoint would be now the one viewing every object proportional to the average visibility multiplied by importance. After incorporating importance, the viewpoint mutual information is given by

$$
I'(v, O) = \sum_o p(o|v) \log \frac{p(o|v)}{p'(o)}, \tag{7}
$$

where

$$
p'(o) = \frac{p(o)i(o)}{\sum_o p(o)i(o)} \tag{8}
$$

and $i(o)$ is the importance of object o.

The importance distribution among the volumetric objects is a direct mapping of the user's interest to a particular part of the volumetric data. Therefore it is very important that the scheme for obtaining the characteristic viewpoint of the object of interest is robust with respect to the importance distribution. The previously used scheme using the VE measure [1], where the probabilities are given by the normalization of the voxel visibility divided by the voxel importance, turned out to be not robust enough for our aims. On the other hand using the VMI measure for obtaining characteristic viewpoints results into much more *stable* results. Figure 4 shows an extreme example of unwanted VE behavior. The test data set consists of a cube divided into two halves, where each half is one object (depicted by dark red and

green colors). The importance is distributed to objects in the following way: the object of interest (*active object*) has importance value 100, the *inactive* object has importance value 1 and the background has importance value 0.1. Visibility has been computed for 6 views, i.e., from each orthogonal view. From these viewpoints most characteristic viewpoints have been computed when no object is active, the dark red half is the active object, and the green half is the active object. The upper row shows the characteristic viewpoints estimated by VE, the bottom row by VMI. The images show that after incorporating importance, VE has in fact selected the wrong viewpoints as the best ones, while VMI performed correctly.
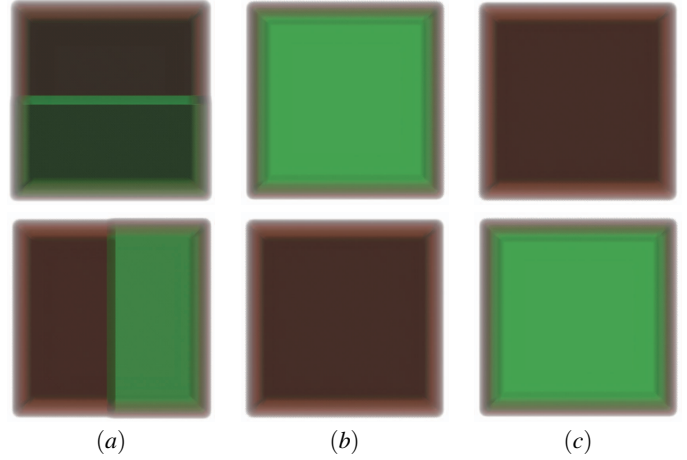

Fig. 4. Behavior of viewpoint entropy (upper row) vs. viewpoint mutual information (bottom row) tested on a cube data set with two halves with different importance. Characteristic viewpoints when (a) no active object selected, (b) red half is active (high importance assigned to dark red object), and (c) green half is active (high importance assigned to green object)).

While VMI takes into account the degree of visibility of the objects seen with respect to all the knowledge about visibilities and importances of all the objects, VE only takes into account the visibility and importance of the objects seen, without any other reference. Thus, as it can be seen in our example, one object (background) with high visibility and very low importance can distort completely the desired results.

On the other hand, while the maximization of VE tries to balance the probabilities obtained from the normalization of the quotients $p(o|v)/i(o)$ of the objects seen [1], the minimization of VMI tries that the visibility of the objects comes close to the normalized product $p(o)i(o)$. For instance, given a view, if we scale the importance of all the objects in the view, VE will not change. This means that two different views with equal visibility distribution and different importance will have the same VE. On the contrary, in this case VMI will select the view where the most important objects are visible.

4.4 Obtaining Characteristic Viewpoints

Equation 7 defines the VMI with importance classification. This is computed for each viewpoint and for each active object separately (as they have different visual representations, which implies different visibilities). To obtain a set of characteristic views for a given object o, we compute the conditional probabilities of all objects for a given viewpoint. The conditional probability $p(o|v)$ is equal to the normalized visibility, i.e., the visibility of all objects per viewpoint are equal to 1 as described in Section 4.2.

Furthermore we have to compute the marginal probability $p(o)$ from Equation 3. To compute $p'(o)$ we first compute a dot product between the marginal probability vector

$(p(o_0), p(o_1), p(o_2), ..., p(o_{m-1}), p(o_m))$ and the importance distribution vector $(i(o_0), i(o_1), i(o_2), ..., i(o_{m-1}), i(o_m))$ where m is the number of objects and o_0 is the background volume. After the sum in the denominator of Equation 8 is computed, all information is available and we can compute the VMI for viewpoint v.

The VMI is computed for every viewpoint and the set of viewpoints with the smallest mutual information are selected. These computations give us good viewpoints for a particular active object. To compute good viewpoints for another object, we have to take another set of visibilities where the visual emphasis is on the respective object. All values necessary for the viewpoint mutual information can be stored in a set of 2D schemes as depicted in Figure 3.

5 INTERACTIVE FOCUS OF ATTENTION

How to obtain characteristic viewpoints has been described in the previous section. Let's assume we have identified a set of most characteristic viewpoints per object under the given sparse and dense visual representations. Now we will describe in detail how the focus of attention can be used for the interactive visual inspection of a feature within the volumetric data. The interactive stage of our framework is depicted in the lower part in Figure 3.

The general idea is to use the importance distribution as a steering parameter for the focus of attention. We specify a high importance value for the active object o_1 (e.g., 100.0). The other objects are assigned a low importance value (e.g., 1.0) and the lowest value is assigned to the background (e.g., 0.1). Such a distribution defines that the viewpoint is located at the characteristic viewpoint of the active object, which has a dense representation while other objects and background are represented more sparsely.

By selecting another object (o_2) to become the active object on the user's request, the importance of the previously selected active object (o_1) starts decreasing to the value of inactive objects (1.0). This causes that the visual representation of object o_1 is becoming more sparse and the viewpoint is moving away from the characteristic viewpoint of o_1 towards the characteristic view of the entire scene (*contextual* view).

When the low importance value of object o_1 is achieved, all objects have equal importance. This means no object is visually emphasized and the viewpoint is set to a contextual view. This view provides the context information of all structures so the user does not lose his orientation within the volume. This way of presenting objects is in the spirit of the navigation on large 2D maps proposed by van Wijk and Nuij [12].

Afterwards the importance of the newly selected active object o_2 is increasing to the maximal value (100.0). The change in the importance distribution gives the visual prominence to the object o_2 and the viewpoint is changing to the characteristic viewpoint of o_2.

This is the basic idea of our interactive focusing approach for guided navigation through objects in volumetric data. We describe the details on viewpoint transformation and changes visual representations in the following subsections separately.

5.1 Viewpoint Transformation

For each object we calculate in the pre-processing step one characteristic viewpoint, and several *contextual* characteristic viewpoints for the entire scene. All these viewpoints are located on a bounding sphere around the volumetric data set. When selecting a new active object (o_2) by the user, the importance distribution starts changing which implies the viewpoint position change.

The viewpoint is changing along the path on the bounding sphere. This path starts at characteristic viewpoint of previously active object o_1 (v_1), visits one contextual characteristic viewpoint (v_c), and ends at the characteristic viewpoint of o_2 (v_2). Thus viewpoint change considers three pre-selected viewpoints. We calculate the path as a Beziér curve defined by viewpoint positions (v_1, v_c, and v_2) as three control points. This means that the contextual view on the whole volume is not visited exactly, it is approximated by similar views that also satisfy the goal of providing context. The Beziér curve among three viewpoints is depicted in Figure 5.
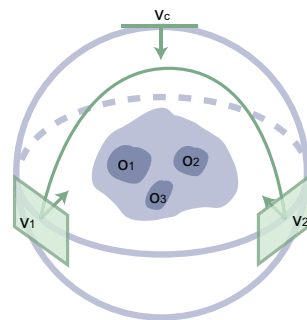

Fig. 5. Change between two characteristic viewpoints of different objects (v_1 for o_1 and v_2 for o_2). The contextual viewpoint v_c is *nearly* visited and is approximated by the Beziér curve.

As we define several characteristic contextual viewpoints (v_{ci}), we have to define which one will be selected as v_c. One possibility can be to select that contextual viewpoint v_{ci} where the viewpoint path will be minimal. This can be done in the following way: For each v_{ci} we compute the angle ϕ_1 between the normal vector of viewpoint v_1 and viewpoint v_{ci} and angle ϕ_2 between the normal vector of v_{ci} and v_2. The contextual viewpoint v_{ci} with the smallest angle sum $\phi_1 + \phi_2$ is selected as v_c. This will guarantee that the overall viewpoint path is the shortest.

The position of the viewpoint always has to be located on the bounding sphere. To obtain the set of intermediate viewpoint positions, the Beziér curve is sampled by equally distant samples. These samples are then projected onto the bounding sphere. These projected samples define the centers of the intermediate viewpoints.

This has one favorable implication: The viewpoint change starts slowly, has the biggest angle difference (i.e., viewpoint change speed) in the middle between the characteristic view v_1 and v_c, where the viewpoint transformation slightly slows down to dedicate a little time to the contextual view. Then it speeds up again and slows down before reaching the characteristic viewpoint v_2. Entire transformation results into visually pleasing viewpoint changes.

This implication is shown in Figure 6. For the sake of clarity the viewpoint change difference is illustrated only by two viewpoints v_1 and v_2 where the viewpoint path is a line projected on the sphere. In the middle of the viewpoint path the change is the fastest, while close to more interesting viewpoints v_1 and v_2 the viewpoint change is slower.
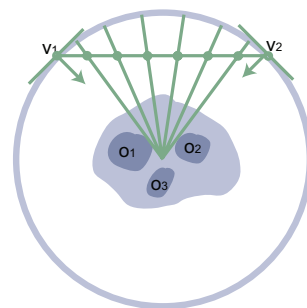

Fig. 6. The viewpoint path is calculated as a difference between two viewpoint positions. The path is then normalized onto the bounding sphere, which smooth acceleration and de-acceleration in viewpoint change.

An important consideration in the viewpoint setting with respect to a visually pleasing focusing, is the orientation of the viewpoint *up-vector*. In our implementation we set the viewpoint up-vector to point towards the up-vector of the volume. The up-vector of the volume is defined for each dataset before the visual inspection. This is done together with all other preprocessing steps: defining objects by seg-

mentation, defining visual representations, and estimating good viewpoints. If the viewpoint is located at the *poles*, i.e., the viewpoint normal vector is parallel to the volume up-vector, we select another vector to be the viewpoint up-vector. In our implementation we use the volume front-vector with inverse orientation so we look at the volume from top-front.

Apart from the guided navigation through interesting objects within the data, our system allows free user manipulation with the viewpoint to see the inspected object from various viewpoints. User has to enable the free manipulation mode. When the free manipulation mode is later disabled by the user, the viewpoint smoothly changes to last selected object of interest.

5.2 Change in Visual Representation

A characteristic view is one important part of focus of attention. However without emphasis through the selected visual representation, the focus object is still not discriminated from the context objects. Therefore in our framework changes in importance distribution also change the visual appearance of objects. A visual representation basically changes in a similar way as the viewpoint. In this case we do not need to calculate a path. We select the appropriate level of sparseness in the visual representation. In our implementation we define the visual representations of inactive and active objects before the visibility calculation. These visual representations can be linearly interpolated for example. In our focusing framework we use a discontinuous change in the visual representation as this abrupt change attracts an observer's attention much stronger. While the viewpoint moves from view v_1 (showing the previously active object o_1) towards the *contextual* view v_c (the importance of the previous active object is decreasing), the previous active object is still visually emphasized. After reaching the context viewpoint, the visual representation of the previously active object is suppressed and the new active object o_2 is visually emphasized (from the moment when importance of o_2 starts increasing).

In addition to changes in the visual representation, we optionally incorporate cut-away views to give a clear view at internal objects. The *level of ghosting* in the cut-away region in front of the interesting feature is also driven by importance changes. In this case we do not employ abrupt changes, but the level of ghosting changes smoothly. This means the ghosting level (i.e., opacity in the cut-away region) is increasing with decreasing importance of the previously active object o_1 and is decreasing with increasing importance of the new active object o_2. When the characteristic view is reached, the ghosting level is minimal, i.e., features in front of the active object are in the cut-away region completely transparent.

We also optionally include additional information into this static view (i.e., when the importance distribution is not changing) by blending-in textual annotations. These are represented in form of overlay labels that indicate the name of the selected object. In case when all objects are labelled the label of the active object is visually more prominent (see Figure 1).

6 Results

We have integrated the focus of attention functionality as a plugin into VolumeShop [2]. This system allows easy prototyping with the possibility of using a lot of existing functionality. We have extended the information about the data set, which is stored in an XML structure, by information on the volume up-vector and on the volume front-vector. After the viewpoint estimation, a characteristic viewpoint is saved for each object in the XML structure as well as the set of global *contextual* views. Visibility computation for each object is the most time-consuming part of the framework and takes approximately a few minutes. Visibility computation through the ray-casting has to be performed for a large number of viewpoints. Currently our implementation takes about 15 seconds per viewpoint for $64 \times 64 \times 64$ data set with two objects plus background on an AMD 64 X2 Dual Core 4800+ in a single thread. This time consumption can be shortened by a more efficient implementation. As this is a pre-processing step that is considerably shorter than object specification by segmentation or setting-up proper visual representations, this is not a real issue. During the

user interaction the performance is approximately 20 to 5 frames per second depending on the data size, and graphics hardware. The graphics hardware has to support the pixel shader PS 3.0 specification. Additional viewpoint location computations as well as importance-driven modifications of visual representations do not take any noticeable time and the performance is equal to standard multi-volume rendering implemented in VolumeShop [2].

Focus of attention is shown on three different data sets. The human hand and torso (Figures 1 and 2) show objects that are inside the data set. In this case the visibility computation used cut-away views to identify the best visibility. Interesting objects are shown from their characteristic viewpoints. More prominent visual representation discriminates the interesting feature from other parts of the data. The visualization is additionally extended by labels showing the names of respective emphasized parts.

In case of the stag beetle data set (Figure 7), only outer parts have been selected so the option for cut-away visibility calculation was not enabled. In this figure sample images have been taken from a viewpoint path that re-focuses from the thorax to the legs. Between the fourth and fifth image the contextual viewpoint has been reached and the focus is switched to the legs.

7 Summary and Conclusions

In this paper we have proposed the concept for importance-driven focus of attention. We have discussed the necessary pre-processing steps before a visual inspection puts the focus of attention on interesting objects. One of these steps is localization of viewpoints that show characteristics of an object in the best way. We use a new method for viewpoint selection for volume data using viewpoint mutual information that works very good for segmented volumetric data classified by importance.

We have shown possibilities how to realize focus of attention for a visual inspection of volumetric data with added information such as varying visual representations, characteristic viewpoints for objects and the entire volume, up-vector of the volume.

We have discussed aspects of a visually pleasing re-focusing from one object of interest to another. This includes the selection of viewpoints, design of a path for the viewpoint and also changes in the visual representation. Inspection of pre-selected structures gives a good overview on the information content of the underlying data.

8 Future Work

This section presents some improvements and extensions that can be incorporated into the presented framework. The current performance of the proof of concept implementation for visibility estimation is quite time consuming. We expect to improve the performance considerably by porting the computation towards GPU.

The viewpoint path is designed by connecting characteristic views of objects and a global characteristic view. This means that the information-theoretic framework does not have an entire control over the path design. With a fast visibility estimation scheme it will be interesting to compute the path design directly using the viewpoint mutual information measure.

At the moment the viewpoint change is realized on the bounding sphere of the entire volume data. To see the inspected object in a more detail, an automatic zooming strategy at the objects of interest can be considered. During re-focusing from one object a zoom-out will be performed and again zoom-in at the newly selected object of interest.

The quality of a viewpoint is hard to justify. In fact only a user study could be used for viewpoint quality evaluation. An interesting point can be to let users determine their own preferred viewpoints. This information can be used as input to the existing IT framework as marginal probability of the viewpoint which is currently set to a constant value for all viewpoints.

Fig. 7. Stag beetle data set: re-focusing from the thorax to the legs.

REFERENCES

[1] U. Bordoloi and H.-W. Shen. View selection for volume rendering. In *Proceedings of IEEE Visualization'05*, pages 487–494, 2005.

[2] S. Bruckner and M. E. Gröller. VolumeShop: An interactive system for direct volume illustration. In *Proceedings of IEEE Visualization'05*, pages 671–678, 2005. http://www.cg.tuwien.ac.at/volumeshop/.

[3] N. Burtnyk, A. Khan, G. Fitzmaurice, and G. Kurtenbach. ShowMotion: camera motion based 3D design review. In *Proceedings of the Interactive 3D Graphics and Games Symposium*, pages 167–174, 2006.

[4] T. Cover and J. Thomas. *Elements of Information Theory*. Wiley Series in Telecommunications, 1991.

[5] B. Gooch, E. Reinhard, C. Moulding, and P. Shirley. Artistic composition for image creation. In *Proceedings of Eurographics Symposium on Rendering '01*, pages 83–88, 2001.

[6] R. Kosara, S. Miksch, and H. Hauser. Semantic depth of field. In *Proceedings of IEEE InfoVis '01*, pages 97–104, 2001.

[7] C. H. Lee, A. Varshney, and D. Jacobs. Mesh saliency. In *Proceedings of ACM SIGGRAPH '05*, pages 659–666, 2005.

[8] M. Sbert, D. Plemenos, M. Feixas, and F. González. Viewpoint quality: Measures and applications. In *Proceedings of Computational Aesthetics '05*, pages 185–192, 2005.

[9] N. Svakhine, D. Ebert, and D. Stredney. Illustration motifs for effective medical volume illustration. *IEEE Computer Graphics and Applications*, 25(3):31–39, 2005.

[10] T. Taerum, M. Sousa, F. Samavati, S. Chan, and J. Mitchell. Real-time super resolution contextual close-up of clinical volumetric data. In *Proceedings of EuroVis '06*, pages 347–354, 2006.

[11] S. Takahashi, I. Fujishiro, Y. Takeshima, and T. Nishita. A feature-driven approach to locating optimal viewpoints for volume visualization. In *Proceedings of IEEE Visualization'05*, pages 495–502, 2005.

[12] J. van Wijk and W. Nuij. A model for smooth viewing and navigation of large 2D information spaces. *IEEE Transactions on Visualization and Computer Graphics*, 10(4):447–458, 2004.

[13] P. P. Vázquez, M. Feixas, M. Sbert, and W. Heidrich. Viewpoint selection using viewpoint entropy. In *Proceedings of VMV '01*, pages 273–280, 2001.

[14] P. P. Vázquez, M. Feixas, M. Sbert, and W. Heidrich. Automatic view selection using viewpoint entropy and its application to image-based modeling. *Computer Graphics Forum*, 22(5):689–700, 2003.

[15] I. Viola, A. Kanitsar, and M. E. Gröller. Importance-driven feature enhancement in volume visualization. *IEEE Transactions on Visualization and Computer Graphics*, 11(4):408–418, 2005.