

# Design Principles for Cutaway Visualization of Geological Models

Endre M. Lidal\*  
University of Bergen, Norway

Helwig Hauser†  
University of Bergen, Norway

Ivan Viola‡  
University of Bergen, Norway  
Christian Michelsen Research, Norway

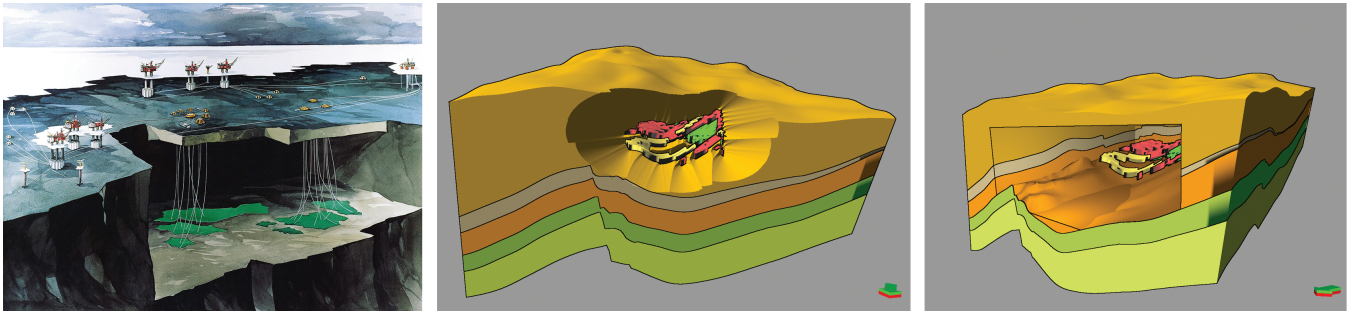


Figure 1: *Left:* A geological cutaway illustration. Image is courtesy of Statoil. *Middle:* Our implementation of the cutaway method by Burns et al. [2008] on the Ness geological model. It exemplifies one of the perceptual problems we observed with current cutaway methods; it looks like the geological channels are located in the yellow-colored top layer, while in fact they are all residing in the orange, third layer. *Right:* A cutaway visualization based on our design principles. In this picture we more clearly see in which layer the geological channels resides, how the channel shapes corresponds to the shape of the surrounding context, and the depth of the cutaway.

## Abstract

In this paper, we present design principles for cutaway visualizations that emphasize shape and depth communication of the focus features and their relation to the context. First, to eliminate cutaway-flatness we argue that the cutaway axis should have an angular offset from the view direction. Second, we recommend creating a box-shaped cutaway. Such a simple cutaway shape allows for easier context extrapolation in the cutaway volume. Third, to improve the relationship between the focus features and the context, we propose to selectively align the cutaway shape to familiar structures in the context. Fourth, we emphasize that the illumination model should effectively communicate the shape and spatial ordering inside the cutaway, through shadowing as well as contouring and other stylized shading models. Finally, we recommend relaxing the view-dependency constraint of the cutaway to improve the depth perception through the motion parallax. We have identified these design principles while developing interactive cutaway visualizations of 3D geological models, inspired by geological illustrations and discussions with the domain illustrators and experts.

**CR Categories:** I.3.5 [Computer Graphics]: Computational Geometry & Object Modeling— [I.3.8]: Computer Graphics— Applications

**Keywords:** Illustrative Visualization, Cutaway Visualization, Visualization in Earth, Space, and Environmental Sciences

\*e-mail: Endre.Lidal@uib.no

†e-mail: Helwig.Hauser@uib.no

‡e-mail: Ivan.Viola@uib.no

## 1 Introduction

Presenting and communicating important features in complex 3D geometric models is challenging. Professional illustrators utilize many different techniques for achieving this aim. The cutaway is a technique often used to expose focus features that otherwise would be occluded. The cutaway metaphor has also been adapted to interactive visualization. In this paper, we argue for the need of new, extended design principles to effectively apply interactive cutaway visualizations to models from domains previously not investigated. Our application-side motivation is to use visualization to communicate important features found inside 3D geological models more clearly. We want to provide the experts from the geoscience domain with an authoring tool that they can use to create interactive visualizations for communicating the important features in the geological models they create.

Traditionally, these models were communicated through hand-crafted illustrations. Computers and simulations have taken over most of the modeling and rendering of geological models. The 3D computer tools that are currently used in the earth science domain have, however, not adopted its illustrative heritage. Thus, many of the expressive techniques used earlier for communicating the models have been lost in this workflow transition [Merriam 2009].

Our study focuses on sedimentary deposit geological models. These models are made, e.g., during ground-water mapping, or oil and gas exploration. Sedimentary models contain a set of geological layers having different geological properties. Structural changes, such as folding or faulting, can change the direction and shape of these layers, but the models often resemble similarity to the cake-layered model seen in most geological illustrations. Typical focus features in a geological model are sand-bodies, i.e., potential water, oil, and/or gas reservoirs, or structural bodies, like salt. The left image in Figure 1 shows an example of geological focus features depicted in a cutaway illustration. Further examples can be found on the Internet [John Perez Graphics & Design 2012]

When we applied known cutaway visualization solutions to geological models, we identified serious depth perceptual problems, as depicted in the middle image in Figure 1. To improve the cutaway

visualization we sought inspiration from geological illustrations, from vision and perception sciences, as well as from discussions with geologists and illustrators. Based on this we have formalized five design principles for creating cutaway visualizations that work well with geological models and other models that share the same characteristics. These design principles, presented in Section 3, include recommendations of relaxing the view-dependency of the cutaway, to use a simple cutaway shape, and to use an illumination model that effectively communicates the shape and spatial ordering inside the cutaway. The right image of Figure 1 shows the result.

## 2 Related work

Cutaway visualizations were utilized in early works on volume data rendering, e.g., by Levoy [1989]. The first work that put cutaway as the main focus was by Feiner and Seligmann [1992]. Another early work that resembles an illustrator's cutaway, is the 3D magic lenses by Viegas et al. [1996]. Diepstraten et al. [2003] later proposed a set of requirements for producing *interactive* cutaway illustrations. The work of Coffin and Höllerer [2006] extended these earlier publications by proposing a user-defined cutaway that adapts to the occluding geometry. Weiskopf et al. [2003] showed how to create both convex and concave interactive cutaways in texture-based volume data visualizations. The publications mentioned above were limited by and specially tuned to the graphics hardware of their time.

Viola et al. [2005] introduced *view-dependent* cutaways for volumetric data and coupled this with the integration of important driven techniques. Their cutaways were data-driven, i.e., the cutaway shape was automatically defined from the focus features. Bruckner and Gröller [2005] made an interactive implementation of this data-driven, view-dependent cutaways. Burns et al. [2008] adapted the data-driven, view-dependent cutaways for polygonal meshes. Their polygonal solution requires double-walled polygons to provide cutaway walls. Kubisch et al. [2010] utilize cutaways to expose tumors for surgery planning. They acknowledge the problem of low depth perception inside the cutaway and suggest using shadows and distance markers on the cutaway wall. All these view-dependent cutaways are image-based and thus highly connected to the viewpoint and the camera position. Li et al. [2007] gave a thorough overview of cutaways in traditional illustrations and presented an advanced system for creating cutaways. Their implementation is based on CSG models and this is not generally applicable to the complex shapes of geological features. Furthermore, they define cutaways primarily for static viewpoints rather than the dynamic cutaways for interactive viewpoints that we aim for. Elmqvist and Tsigas [2008] have created a taxonomy of occlusion management for visualization. They rate the cutaway visualization as *very low* quality when depth cues are considered.

Only a few works on cutaways specifically targeting the earth science domain has been presented. Ropinski et al. [2006] introduced a view-independent cutaway for seismic datasets and this work was followed up by the roaming box, also for seismic datasets, by Patel et al. [2007]. The focus in both of these papers is on the search process of exploring a dataset of measured samples. This process usually has different requirements and solutions than the visual communication process we target here in our work. Turner [2005] presented general characteristics of geological models. This work is especially interesting because Turner uses these characteristics to explain the challenges of building computer representations and performing the visualization of geological models.

In the recent book by Thompson et al. [2011] they provide a thorough overview of works and theory from vision science and visual

perception in the context of computer graphics. Our design principles are rooted in these sciences.

The 3D geological models we visualize consist of a set of sealed geometric objects defined as polygon surfaces, often used in geological models [Caumon et al. 2004]. Thus, the cutaway algorithms for volumetric data mentioned above are not applicable, unless they can be adapted to polygonal scenes. Burns' adapting of Viola's work is one such example. The cutaway algorithms that seem most applicable to our needs are: Burns et al. [2008], Li et al. [2007] and Kubisch et al. [2010]. We compare these works to our design principles in the next section.

## 3 Geological Model Characteristics and the Design Principles

We observed low depth perception when we applied known cutaway visualization methods to 3D geological models. This low depth perception hampered the model understanding in the following ways:

- difficulties to perceive the spatial ordering of focus features, i.e., the ordinal distance between focus objects
- difficulties to understand the relative distance of focus features to the cutaway walls, i.e., the position of the focus features inside the cutaway
- difficulties to perceive the shape and topology of the focus features
- difficulties to understand the spatial relationship between the focus features and the context, e.g., within which geological layer is a focus feature residing
- difficulties to understand the relationship between the cutaway shape and the model

To understand why the known cutaway approaches failed for geological models we turned to Turner [2005]. It is common to model the subsurface a set of geological features or units, called *geo-objects*. Turner argues that these geo-objects have characteristics that pose big challenge for the modeling and the visualization of geology, especially compared to visualization of engineering and other man-made objects. He lists, among others, these characteristics: *complex geometry and topology* and *indistinct boundaries defined by complex spatial variations* [Turner 2005], characteristics we believe are the main reasons why current cutaway visualizations fail. Turner points out that a tool for geology must be capable of efficiently model and visualize: *Geometry of rock- and time-stratigraphic units*, and *spatial and temporal relationships between geo-objects*. These requirements target what we listed as shortcomings of today's computer-generated cutaway visualizations.

According to Thomson et al. [2011], *familiarity* with the object's shape contributes to the scene perception. A major difference between the geo-objects and the objects from, for instance, medical illustrations is that of *familiarity*. Sand-bodies, folded horizons, and faults, which result from complex geo-physical processes, vary significantly in structural arrangement from one geological model to another. Therefore, the geological cutaway visualization cannot rely on the viewer's familiarity with a specific model. In addition, in medical illustration cutaways are predominantly utilized for an *onion-like* structural arrangement. Geological structures have usually a *layer-cake* structure. This difference is an additional challenge in the cutaway design.

With the knowledge from vision and perceptual science, the geo-object characteristics, and the geology visualization requirements

in mind, we studied geological illustrations and had discussions with geology illustrators and geology domain experts. From this, we made the following observations and derived design principles for computer-generated illustrative cutaway visualization.

**Design Principle 1:** *Use an oblique viewpoint to the cutaway.* Studying cutaway illustrations like in Figure 1 and the examples found online at [John Perez Graphics & Design 2012], we observed that the viewpoint is rarely ever positioned so that the viewer is looking directly into the cutaway. Instead, the illustrations all have an oblique viewpoint to the cutaway, mostly slanted from the left side. This oblique viewpoint increases the depth perception by providing a frame of reference (the cutaway wall) for perceiving the depth. This makes the cutaway look less flat (more 3-dimensional), analogous to the way how the shape perception of an object is increased by moving a light source from a headlight position to an “over-the-shoulder” position. Note that the oblique cutaway is something different than beveling, proposed by other cutaway methods, such as [Li et al. 2007]. Beveling (the cut angle) exposes the cross sectional surface of the cut, providing better depth understanding of the cut surface. The slanted view in oblique cutaway, however, provides visual cues of the depth relations between the focus feature and the context. Oblique cutaway can be combined with beveling.

**Design Principle 2:** *Simple cutaway shape for complex models.* Because of the complexity in the geological models the illustrators draw the cutaway as a simple shape. They often use a rectangular-based box shape, sometimes similar to the overall shape of the geological model, also pointed out by [Li et al. 2007]. A simple and easy-to-understand cutaway shape makes it easy to extrapolate the complete model and to understand how deep into the model the cutaway extends. As a general advice, we recommend a rectangular box-shaped cutaway, like geology illustrators often use.

**Design Principle 3:** *Include familiar context.* Important relationship aspects between focus features and context are often expressed by keeping some parts of the context, close to the focus features, uncut. The artist can deliberately leave some minor parts of the focus features occluded to emphasize the relationship between the focus and the context. This can be seen in the examples referenced online [John Perez Graphics & Design 2012].

**Design Principle 4:** *Utilize an illumination model that effectively communicates shape, and ordinal and relative distances.* We observed that the illustrations often communicate rock or stratigraphic properties through exaggerated rendering techniques and textures. The illustrators also draw shadows between the focus features and the context to emphasize spatial distance and spatial ordering.

**Design Principle 5:** *Take advantage of the motion parallax depth cue.* Although we studied mostly static illustrations, it is known to vision science [Thompson et al. 2011] that motion and interaction are mechanisms that can increase depth and shape understanding of 3D models. Thus, it is important to also take advantage of interaction and movement in the computer-generated cutaway visualizations.

Comparing the design principles to previous works, we get:

**Burns et al.:** Their cutaway supports neither the oblique cutaway nor motion parallax depth cue. The cutaway shape is too complex for geological model and it is not possible to include context features in the cutaway.

**Kubish et al.:** Their work acknowledges the depth perception issues inside the cutaway and they suggest some illumination techniques to improve this. It does not support oblique cutaways, motion parallax depth cue, or simple cutaway shape needed for geological models.

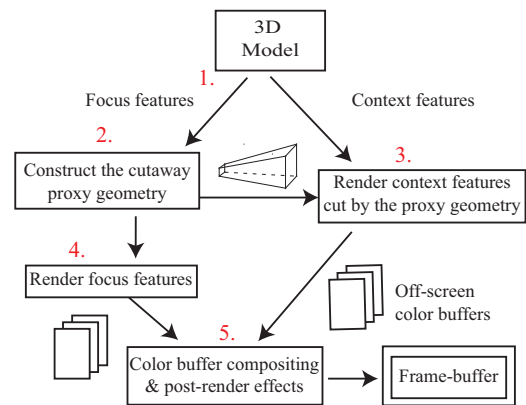


Figure 2: The proposed cutaway visualization pipeline.

**Li et al.:** Their work is strong on suggestions for lighting models for enhancing shape and topology of cut objects and it supports linking the focus features with the context. However, it requires predefined viewpoints and lacks support for oblique cutaways.

Even though some of the current cutaway algorithms cover some of the design principles, none of them covers all of the principles.

## 4 The Design Principles Realized

The input model to our cutaway visualization is a set of 3D polygonal objects. Our solution uses a *cutaway proxy geometry* for defining the *cutaway-volume*. This proxy geometry has initially the shape of a truncated rectangle-based pyramid with parallel front and back planes. The individual steps of our cutaway visualization solution are described below and also illustrated in Figure 2:

1. The polygon objects of the model are divided into two groups, one or more *focus features* and the *context*.
2. The cutaway proxy geometry is constructed based on the depth footprint of the focus features.
3. The context objects of the geological model are intersected by the cutaway proxy geometry and the cutaway walls are constructed. For each context object, the cutaway walls and the parts outside the cut volume are rendered to a separate off-screen color buffer. By rendering each feature to separate off-screen buffers, we can add compositing effects when synthesizing the final output image.
4. The focus features of the model are each rendered into separate off-screen color buffers.
5. After all objects are rendered, the off-screen color buffers are composited into one buffer. Compositing effects, such as silhouette rendering, can be applied in this step. In this final step, before copying the image to the frame-buffer, we can also add perception enhancing screen-space post-rendering effects.

In this section we describe in detail how to realize the design principles.

### 4.1 Oblique Cutaway (DP1)

Thompson et al. [2011] writes that a spatial frame of reference is needed to judge shape and distance. For a computer-generated cutaways this reference frame can be provided by drawing the cutaway

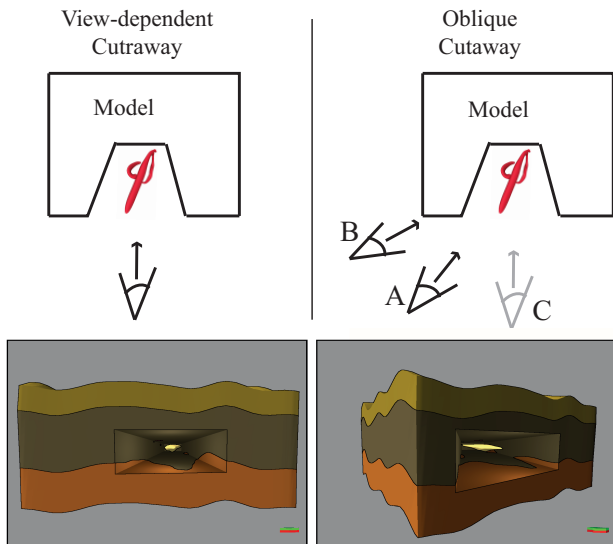


Figure 3: *Top left*: A schematic top view of a cutaway visualization where the camera points directly into the cutaway. *Top right*: The same cutaway, but with an oblique viewpoint. The cutaway proxy geometry is generated from the camera position C and the scene is rendered from the camera position A. Note: with large angles (camera positions B), the focus feature is occluded by the context. *Bottom*: The corresponding cutaway visualizations.

with an oblique viewpoint. Provided this oblique viewpoint the observer can perceive the shape and depth of the focus objects against the side cutaway walls, something that is difficult with a direct-on cutaway viewpoint.

#### 4.1.1 Constructing the Cutaway Proxy Geometry

To create the oblique cutaway we need to relax the tight coupling between the cutaway and viewing direction found in view-dependent cutaways. We apply one camera transformation to create the cutaway and another transformation to render the scene, see the illustration in Figure 3. Because of the change in view transformation we must create a polygonal model defining the cutaway volume, i.e., a *cutaway proxy geometry*, and this proxy geometry must be defined in model-space.

The proxy geometry we propose is initially frustum shaped, defined by 8 corner points. It is illustrated in Figure 4. The steps to define these 8 points are:

1. Render the depth footprint of the focus features to an off-screen depth buffer, (see (1) in Figure 4).
2. Extract the maximum depth value and the bounding box around the depth footprint, (see (2) in Figure 4).
3. Transform corners to view coordinate space, (see (3) in Figure 4).
4. Define the front plane depth value by finding the point in the model closest to the camera and then use its depth value, with a small offset, for the depth distance of the front plane.
5. The corners of the front plane of the cutaway frustum are calculated using the bevel angles, i.e., the slope of the sidewalls in the frustum. It is possible to specify individual bevel angles for each sidewall.
6. After all 8 corners are calculated, they are transformed to the model coordinate space. This is to enable the decoupling from the camera.

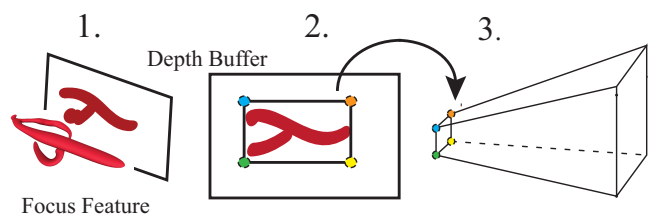


Figure 4: The back plane of the cutaway proxy geometry is constructed from the bounding rectangle around the depth footprint of the focus features (1) – (3)

This algorithm is efficient enough to allow the recalculation of the cutaway proxy geometry whenever needed.

#### 4.1.2 Creating the oblique cutaway

With the proxy geometry defined, cutting the context objects is done in the following three steps:

##### 1) Extract the front and back depth cutaway surface

Current cutaway visualization solutions, such as Burns et al. [2008] and Kubisch et al. [2010], use only one (back) depth cutaway surface for intersecting the context. This makes it impossible to decouple the cutaway from the viewpoint as required by an oblique viewpoint cutaway, because all context in front is cut away. We introduce an additional (front) depth cutaway surface. With both a front and back cutaway surface, we ensure that only the context residing *inside* the cutaway volume are cut. The context outside, that could potentially occlude the focus features, are left uncut, see Figure 5.

Current graphics hardware does not support multiple depth buffers, thus we have to render to two separate off-screen buffers and perform the depth test in the fragment shader. Figure 6 illustrates the front and back depth buffers according to the following algorithm:

1. One off-screen buffer is attached as a depth buffer attachment to a frame buffer object (FBO) and initialized to minimum depth value (0.0). Depth test is enabled, with depth-function “greater”. As the cutaway proxy geometry then is rendered, the surfaces of the geometry furthest away are recorded in the depth buffer. This is the back depth cutaway surface. We also render a diffuse light-map to a separate off-screen buffer during this step. This light-map is later used for shading the cutaway walls.
2. A second off-screen buffer is attached as a depth buffer attachment to an FBO and initialized to maximum depth value (1.0). Depth test is enabled, now with depth-function “less”, thus this time the closest surfaces of the geometry are stored in the depth buffer. This is the front depth cutaway surface.

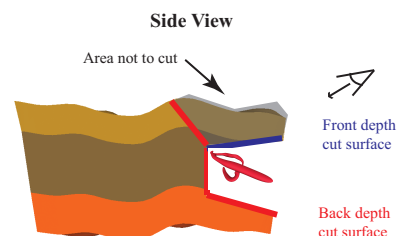


Figure 5: Side view of a cutaway visualization with a vertical angular offset. To preserve the semi-occluding area between the cutaway and the camera, we introduce the front (in blue) cutaway surface in addition to the back (in red) cutaway surface.



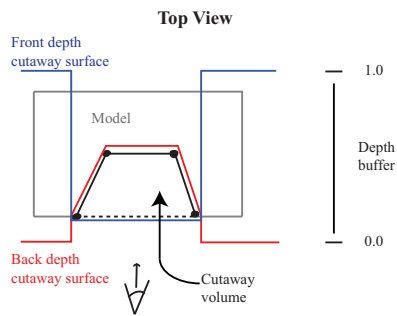


Figure 6: The cutaway volume is the space enclosing depth values that are larger than the front depth cutaway surface and smaller than the back depth cutaway surface.

## 2) Rendering the cutaway walls

Similar to the other cutaway visualizations, we generate cutaway walls to produce the illusion of cutting into solid materials. The cutaway walls also provide a frame of reference for perceiving the depth inside the cutaway. We modified the stencil-buffer algorithm that Kubisch et al. [2010] presented and which they adapted from Coffin and Höllner [2006]. This algorithm renders the context objects correctly without any prior sorting. For each fragment of each context object, we perform a stencil test to see whether the back cutaway surface is inside this context object. If the stencil test passes, we increase the stencil value each time a fragment from a front-facing polygon is rendered and decrease the value for each time a fragment from a back-facing polygon is rendered. We compute this for every fragment that is inside the cutaway volume. With modern graphics hardware, both tests can be done in one rendering pass.

After all polygons of a context object are rendered and if the stencil value is larger than the initial value, then we can conclude that the cutaway surface is inside this context object. We render the fragment to an off-screen color buffer. To resolve the case with context objects completely enclosed by other context objects we use a depth buffer with the depth-test set to "greater". This way we ensure that the fragment color of the context objects closest to the cutaway surface is the final cutaway wall color. The fragment colors are the diffuse material colors of the context object, shaded with the diffuse-light-map generated during the back depth cut surface extraction step.

The stencil buffer is cleared and the depth buffer containing the "greater" depth values is kept between each context object. In addition, we maintain a second depth buffer that stores all the depth values at the cutaway surface for every fragment written. The main difference between our algorithm and the one described by Kubisch et al. [2010], is that we intersect with both a front and back cutaway depth buffer, we render to a set of off-screen color buffers, and we maintain a second depth buffer. Kubisch et al. calculate this second depth-buffer in a separate processing step.

## 3) Rendering the context outside the cut volume

With the depth buffer generated during cutaway wall construction and the front and back depth cutaway buffers, we render the context object again. This time we are discarding fragments that either are inside the cut volume defined by the front and back depth cutaway buffers or further away from the accumulated depth buffer. The result is a set of off-screen color buffers, one for each context object.

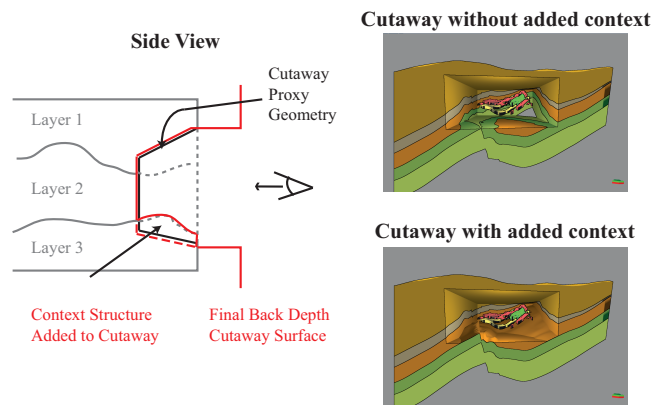


Figure 7: Adding structural parts from the context to the cutaway can be done during back cutaway depth surface generation. Compare the new back cutaway surface (solid red line) with the cut surface without the context structure added (dashed red line).

## 4.2 Cutaway shape (DP2 and DP3)

The frustum shaped cutaway volume utilized so far conforms to DP2, i.e., a simple cutaway shape makes it easy to extrapolate the complete model and to understand how deep into the model the cutaway extends. With DP3, we argue that adding structures from the context to the cutaway volume can improve the understanding of the relationship between the focus features and the context. In addition, identifiable context structures in the cutaway can assist the viewer to understand where in the model the cutaway is made. The context structures are added to the back cutaway surface as part of the preparation step before cutting the context: After we have rendered the cutaway proxy geometry to create the back cut surface, then we render the context structures into the same FBO, now with depth-test set to "nearest". Figure 7 illustrates how the original back depth cutaway surface (in blue) is updated after the context object is rendered (in red). We recommend adding the horizon interface behind the focus features as context structures for the geological models we study, as shown in the result section.

## 4.3 Illumination models (DP4)

Much research has been conducted on shape depiction and inter-object interaction in general computer graphics, but much less attention is paid to objects inside cutaways. The exception is the work by Li et al. [2007]. We emphasize that the illumination model applied when rendering the scene should effectively communicate the shape and spatial ordering inside the cutaway, as well as enhancing relationships between the focus features and the context. Some example of enhancing effects that can be realized during the rendering of the scene objects:

**Context Connectivity Projection** – To emphasize the relationship between the focus features and the context we propose to color parts of the focus features in the same color as the context object it relates to. For geological models, this relationship can for example indicate which geological layer (context) a channel (focus) belongs to. We implement the context connectivity projection as patterns painted on the surface of the focus features in the fragment shader used for the focus feature rendering. Surface color is based on the interpolated model-space coordinate of the fragment. If we consider only one dimension of this coordinate, stripes are produced, for two or three dimensions dots are produced. The user can adjust the frequency and the size of them. See Figure 12 for a demonstration of this effect.

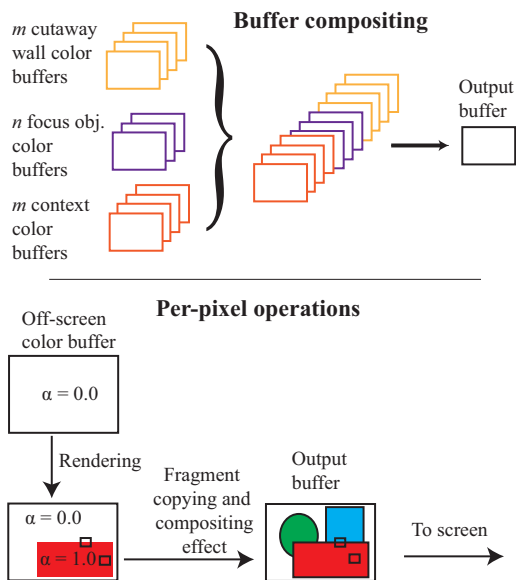


Figure 8: *Top*: The compositing of the off-screen color buffers. No object visibility management is required if the buffers are composited in the order shown. *Bottom*: Compositing is a per-pixel operation, one off-screen color buffer at the time. Only the foreground pixels (alpha value of 1.0) from the color buffers are copied into the output buffer, with possible compositing effects, such as in this example silhouette detection.

**Lighting** – Carefully set lighting can enhance shape communication of an object. To communicate the shape and depth of the cutaway walls better we propose to set a point-light source in the opening of the cutaway when rendering the cutaway walls. Adjusting the attenuation factor of this light source enhances the impression of depth. This lighting setup is used on all images of the conceptual model in this paper. To avoid ending up with flat lighting on the focus features, we utilize the lighting from the upper-left hand side when rendering them.

Compositing is a per-pixel operation of copying pixels from the off-screen color buffers to the output color buffer, one color buffer at the time. Only foreground pixels are copied, i.e., the pixels in the color buffers to which the rendering pipeline actually rendered to. The compositing algorithm identifies foreground pixels through the value of the alpha byte of the pixels, as shown in the bottom image in Figure 8. The stencil buffer and the accumulated depth buffer utilization when rendering the context and the focus features ensures that only the visible parts are rendered to the off-screen color buffers. This simplifies the compositing of the buffers, as there is no need for visibility management. The only requirement is that the buffer sets are processed in the order presented above. The output buffer can be sent directly to the frame-buffer for screen rendering or we can apply screen-space post-rendering effects. Examples of compositing and post-rendering effects are:

**Silhouette Outlines** – Silhouette outlines are important for communicating the shape of an object and to separate overlapping objects. Isenberg et al. [2003] give a good discussion of silhouette extraction and rendering techniques. In our implementation, we apply a Sobel-filter that detects the edges between foreground and background in the color buffer while processing the fragments during the merging of an off-screen buffer. If the fragment is on an edge, we draw the fragment in the silhouette outline color (most often black).

**Shadows** – Illustrators often use shadows to emphasize the distance between two overlapping objects or between an object and the back-

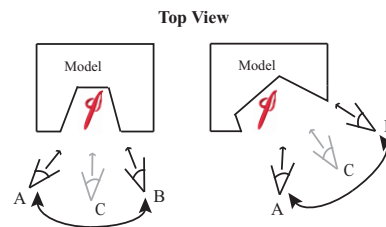


Figure 9: Illustrations of the threshold-based locked cutaways. *Left*: The camera can move between positions (A) and (B) without triggering recreation of the cutaway proxy geometry. *Right*: If the camera moves outside of these thresholds the cutaway automatically follows the camera until the camera moves in the opposite direction again and new thresholds are set.

ground. Similar to Li et al. [2007], we also use the unsharp-masking technique by Luft et al. [2006]. This is shown in Figure 12. In our implementation, we apply this effect on the output results from the compositing step.

#### 4.4 Motion Parallax Depth Cue (DP5)

The effect of the motion parallax implies that when in motion, objects closer to the viewer seem to move faster than objects further away. This perceived difference in angular velocity can be used to communicate the spatial distance between objects. To take advantage of the motion parallax for cutaway visualizations we need to compare the angular velocity of the focus features against a fixed spatial frame of reference, the cutaway walls. Thus, the cutaway must be decoupled from the camera movements and not move when the camera moves. For the view-dependent cutaway visualizations, the cutaway will always moves together with the viewer, thus, the cutaway provides no fixed spatial frame of reference and the depth from motion is not well perceivable.

The cutaway proxy geometry is used to create the motion parallax effect. By completely decoupling the cutaway from the camera movements, we freeze the cutaway at a fixed location in the model. When rotating the camera, this fixed cutaway then provides the needed frame of reference. Completely decoupling the cutaway and camera can result in total focus feature occlusion if the camera is rotated too far, see camera position B in right image in Figure 3. This is why we propose a threshold-based fixed frame of reference cutaway. The user defines two angular threshold values and if the camera rotations are within these threshold values, the cutaway proxy geometry is not updated. When the camera rotates beyond these angular thresholds, the cutaway starts to follow the camera like the oblique viewpoint cutaway described in the previous subsection. This will go on until the camera is rotated in another direction, then the cutaway is fixed and the motion parallax is again more visible. Figure 9 illustrates the threshold cutaway.

## 5 Results and Discussion

Our cutaway visualization solution is implemented in the VolumeShop framework [Bruckner and Gröller 2005], because this framework has good encapsulations of OpenGL functions and good polygon-mesh rendering capabilities.

Our geological partners provided us with a geological model from the Ness-formation of the Oseberg offshore-oilfield in the North Sea. The focus features in this model are sand-filled channels. These channels represent ancient rivers and deltas with excellent

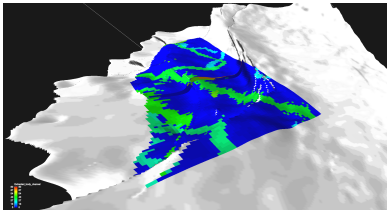


Figure 10: The picture shows how the channels from the Ness-formation are depicted together with the context in the domain specific application called Roxar Reservoir Management Software (RMS).

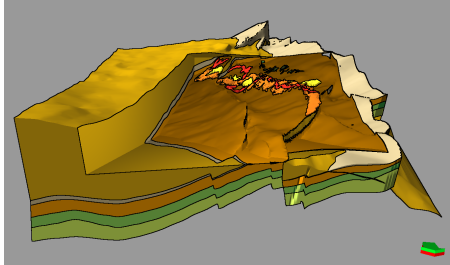


Figure 11: The picture shows the full-field cutaway visualization of the geological model of the Ness formation.

reservoir properties, thus with high possibilities of containing oil and gas. Note that the voxelized shapes of the channels are caused by the conversion from a reservoir simulation grid. We wanted to preserve this shape in our visualization to keep it similar to the way the domain experts visualize them. Figure 10, taken from the petroleum application Roxar RMS, shows a typical visualization of these channels. In order to study the channels together with the context, they are projected down to the underlying geological layer surface, thus losing the spatial ordering and the actual position.

First we selected three of the main channels as focus features and visualized them in the context of the complete Ness formation geology model. The visualization is shown in Figure 11.

Moving to a viewpoint closer to the channels, like the one applied in Figure 12, shows the advantage of adding parts of the context in order to understand the channels better. See for instance in the far end of the picture how it becomes clear why the discontinuities in all three channels are elevated above the main channel parts. Despite the large cutaway of the context in this model, the mapping of context layer color onto the channel (the orange spots) efficiently communicates the relationship between the channels and the geological layer they reside in.

Then we cropped the Ness model around some potentially interesting channel features. This visualization is shown in the right image of Figure 1. One can see the perceptual improvement given by using the simple cutaway shape and the added surface layer below the channels (added context information). For instance, it is easier in this picture to perceive which geological layer the channels reside in. This figure also demonstrates the oblique viewpoint cutaway. Here the depth of the cutaway, i.e., the relative distance from the front of the geological model to the focus features, is clearly communicated.

We also obtained the coordinates of one of the wells in the Oseberg Field and this well is included in Figure 13. Note that this is a different crop-out section of the Ness formation than in the previous images and that the well also continues down to formations below the Ness formation. We think this image demonstrates a big potential for cutaway visualizations during well path planning and

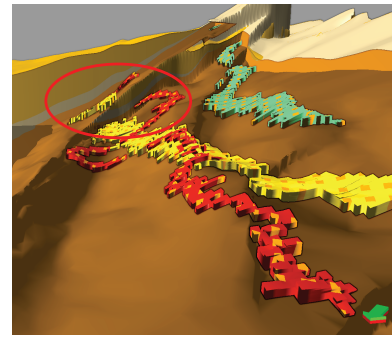


Figure 12: In this close-up of the channels, we see the advantage of adding parts of the context, in this case the layer below the channels, to the cutaway shape. This added context explains why there is a discontinuity and displacement in the far end of the channels (highlighted in red in picture). We are also rendering the color of the context layer as spots on the channels to emphasize the context relationships.

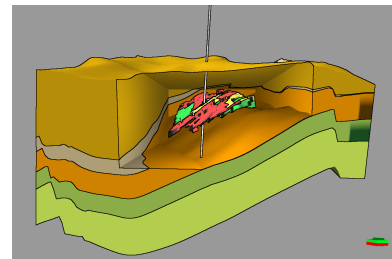


Figure 13: The picture shows a cutaway visualization together with one of the wells from the Oseberg field. This well penetrates the Ness formation before it continues down to other reservoir formations.

verification, which we want to explore in the future.

We have shown our cutaway visualizations to the domain experts, including two geologists and two other experts from our collaborating company in the oil and gas industry, and three PhD students from the Department of Earth Science at our university. They have all confirmed the increased perceptual improvements in the cutaway visualizations that follow our design principles. They also emphasized the importance of seeing the focus features in relationship with the context. Therefore, they especially appreciated that the cutaway shape can include identifiable context and the projection of context layer color onto the channels.

## 6 Conclusion and Future Work

With this work, we wanted to provide new, expert-authored, interactive visualization of the features found inside 3D geological models. We have shown that the previous computer-generated cutaway visualization methods have limitations when attempting to communicate the focus features visually. Reasons for these problems lie in the complex shape of the features of these models. Based on knowledge from the vision and perception science, inspiration from geological illustrations, and interviews with geological illustrators and domain experts we have formulated five design principles for computer-generated cutaway visualization. We believe that these design principles increase the perception of models from the geological domain when using cutaways as occlusion management method, similar to the illustrators' approach. Domain experts have also confirmed the improvements we claim.

A natural extension to our work is to conduct a larger user study to

quantify the benefits and identify the optimal applications of cutaway visualization for models from the geological domain. This will also include adding more geological structure types in the visualization. For instance, we believe that faults could be used to contribute to the cutaway shape. Another possible extension is to perform a systematic analysis of geological work-tasks and types of geological models, and then create a system to automatically select the better depth and shape enhancing cues. This would be similar to the work that Cipiloglu et al. [2010] have done for general 3D scene rendering.

## Acknowledgements

This work is a part of the GeoIllustrator research project, funded by the Statoil Academia Agreement and by the Petromaks program of the Norwegian Research Council (#200512). We would like to thank Statoil for the geological models, the collaboration, and the valuable feedback. We thank our collaborative partners at the Department of Earth Science at University of Bergen and our colleagues in the Visualization Group at University of Bergen. We acknowledge the developers of the MeshLab tool [MeshLab 2012] and the VolumeShop framework [Bruckner and Gröller 2005].

## References

- BRUCKNER, S., AND GRÖLLER, M. 2005. Volumeshop: an interactive system for direct volume illustration. In *Visualization, 2005. VIS 05. IEEE*, 671–678.
- BURNS, M., AND FINKELSTEIN, A. 2008. Adaptive cutaways for comprehensible rendering of polygonal scenes. *ACM Transactions on Graphics (TOG)* 27, 5, 154.
- CAUMON, G., LEPAGE, F., SWORD, C. H., AND MALLET, J.-L. 2004. Building and Editing a Sealed Geological Model. *Mathematical Geology* 36, 4, 405–424.
- CIPILOGLU, Z., BULBUL, A., AND CAPIN, T. 2010. A framework for enhancing depth perception in computer graphics. In *Proceedings of the 7th Symposium on Applied Perception in Graphics and Visualization*, ACM, vol. 1, 141–148.
- COFFIN, C., AND HÖLLERER, T. 2006. Interactive perspective cut-away views for general 3d scenes. In *Proc. IEEE Symp. 3D User Interfaces (3DUI 06), IEEE CS, Press*, 25–28.
- DIEPSTRATEN, J., WEISKOPF, D., AND ERTL, T. 2003. Interactive Cutaway Illustrations. *Computer Graphics Forum* 22, 3 (Sept.), 523–532.
- ELMQVIST, N., AND TSIGAS, P. 2008. A taxonomy of 3d occlusion management for visualization. *Visualization and Computer Graphics, IEEE Transactions on* 14, 5 (sept.-oct.), 1095–1109.
- FEINER, S. K., AND SELIGMANN, D. D. 1992. Cutaways and ghosting: satisfying visibility constraints in dynamic 3D illustrations. *The Visual Computer* 8, 5-6 (Sept.), 292–302.
- ISENBERG, T., FREUDENBERG, B., HALPER, N., SCHLECHTWEG, S., AND STROTHOTTE, T. 2003. A developer's guide to silhouette algorithms for polygonal models. *Computer Graphics and Applications, IEEE* 23, 4, 28–37.
- JOHN PEREZ GRAPHICS & DESIGN, 2012. SD3: [http://www.geoart.com/portfolio/gra\\_ons\\_04.jpg](http://www.geoart.com/portfolio/gra_ons_04.jpg) and STR2: [http://www.geoart.com/portfolio/gra\\_ons\\_26.jpg](http://www.geoart.com/portfolio/gra_ons_26.jpg), Accessed: February 2012.
- KUBISCH, C., TIETJEN, C., AND PREIM, B. 2010. GPU-based smart visibility techniques for tumor surgery planning. *International journal of computer assisted radiology and surgery* (Apr.).
- LEVOY, M. 1989. Design for a real-time high-quality volume rendering workstation. *Proceedings of the 1989 Workshop on Volume visualization - VVS '89*, 85–92.
- LI, W., RITTER, L., AGRAWALA, M., CURLESS, B., AND SALESIN, D. 2007. Interactive cutaway illustrations of complex 3D models. In *ACM SIGGRAPH 2007 papers*, ACM, vol. 26, 31.
- LUFT, T., COLDITZ, C., AND DEUSSEN, O. 2006. Image enhancement by unsharp masking the depth buffer. *ACM Transactions on Graphics* 25, 3 (July), 1206.
- MERRIAM, D. 2009. Special Commentary A LOST ART: Geological Illustrations. *GSA TODAY* 19, 11, 30–34.
- MESHLAB, 2012. Meshlab, <http://meshlab.sourceforge.net/>, Accessed: February 2012.
- PATEL, D., GIERTSEN, C., THURMOND, J., AND GRÖLLER, M. 2007. Illustrative rendering of seismic data. In *Proceeding of vision modeling and visualization*, 13–22.
- ROPINSKI, T., STEINICKE, F., AND HINRICHS, K. 2006. Visual exploration of seismic volume datasets. *Journal proceedings of WSCG* 6, 1, 73–80.
- THOMPSON, W., FLEMING, R., CREEM-REGEHR, S., AND STEFANUCCI, J. K. 2011. *Visual Perception from a Computer Graphics Perspective*. AK Peters.
- TURNER, A. K. 2005. Challenges and trends for geological modelling and visualisation. *Bulletin of Engineering Geology and the Environment* 65, 2 (May), 109–127.
- VIEGA, J., CONWAY, M., WILLIAMS, G., AND PAUSCH, R. 1996. 3D magic lenses. In *Proceedings of the 9th annual ACM symposium on User interface software and technology*, ACM, New York, New York, USA, 51–58.
- VIOLA, I., KANITSAR, A., AND GRÖLLER, M. 2005. Importance-driven feature enhancement in volume visualization. *IEEE Transactions on Visualization and Computer Graphics* 11, 4, 408418.
- WEISKOPF, D., ENGEL, K., AND ERTL, T. 2003. Interactive clipping techniques for texture-based volume visualization and volume shading. *IEEE Transactions on Visualization and Computer Graphics* 9, 3 (July), 298–312.