Visibility Widgets: Managing Occlusion of Quantitative Data in 3D Terrain Visualization

Martin Röhlig
Institute for Computer Science
Albert-Einstein-Straße 22
18059 Rostock, Germany
martin.roehlig@uni-rostock.de

Heidrun Schumann
Institute for Computer Science
Albert-Einstein-Straße 22
18059 Rostock, Germany
heidrun.schumann@uni-rostock.de

ABSTRACT

In 3D visualization of terrain, occlusion has detrimental impact on discovering, accessing, and spatially relating embedded data. This can lead to significant loss of information. To address this issue, we present visibility widgets: compact auxiliary views that visualize occluded data, their locations, and related properties, and thus, support users in revealing hidden information. Our widgets comprise different occlusion management strategies for detecting and analyzing invisible data inside as well as outside of the field of view. In addition, each widget is designed complementary and transient to facilitate interactive exploration and navigation while not monopolizing the user's attention. We demonstrate the application of our visibility widgets in a flexible visualization tool, focusing on visual exploration of hazardous weather phenomena in 3D terrain.

CCS Concepts

•**Human-centered computing** → **Information visualization;**

Keywords

Data Visualization in 3D Terrain; Occlusion Management; 3D Focus and Detail Views

1. INTRODUCTION

Visually analyzing geo-spatial data in virtual environments is a necessary task in many domains. Prominent examples include wayfinding in digital terrain models [2], assessing photovoltaic potential and residential quality in virtual city models [14], or investigating oceanographic data [13, 18]. Analysts typically have to explore the virtual environments to *discover* and to *access* relevant information, and to *spatially relate* it to the geometry of the frame of reference. In this context, 3D visualization allows for faithful representations of the geo-spatial data and the reference geometry, faciliates orientation via landmarks, and enables analysts to recognize the virtual surroundings in the real world [3]. At the same time, analysis tasks are often severely hampered by occlusion effects inherent to 3D environments [6].

In case of digital terrain models, geo-spatial data are often visualized by mapping data values onto the terrain's surface. Especially for quantitative data, such as meteorological scalar fields for precipitation, wind speed, or temperature, this is a common visualization approach. It allows users to relate data values directly to topological features. Yet, it also implies that relevant information gets partly or fully occluded by the geometry of the terrain. These problems are most prevalent in near-surface perspectives of the environment. As a result, analysts may have difficulties detecting, identifying, and distinguishing partly occluded data. If analysts are even unaware of the existence of hidden information, e.g., because it is fully occluded by a mountain range, they might draw wrong conclusions about the data. Particularly in security critical scenarios, such as analyzing hazardous weather phenomena, this can have major implications.

Several approaches have been developed for managing occlusion. For instance, multi-perspective panoramas have been successfully applied for navigating, wayfinding, and investigating objects in virtual 3D city and landscape models [16]. However, open issues are how to manage occlusion of quantitative data, to efficiently compute and visualize the visibility of such information, and in this way, to support exploration of information-rich geo-virtual environments. To close this gap, we introduce visibility widgets: compact auxiliary views that facilitate revealing and analyzing partly or fully occluded information, inside as well as outside of the field of view. Our contributions are:

Detecting Occluded Data: We identify different types of occlusion for quantitative data in 3D terrain. We discuss their computation in general and present efficient implementations on graphics hardware to support visual-interactive exploration of complex environments.

Presenting Occluded Data: We visualize hidden information via novel visual designs instantiated by four complementary visibility widgets. Each widget comprises different occlusion management strategies to (i) overview visible and occluded data, and to (ii) locate and analyze that data in a main 3D visualization.

Interacting with Occluded Data: We provide coordinated interaction techniques for (i) selecting and highlighting information in interlinked views, (ii) investigating details on-demand, and (iii) facilitating discovery, access, and spatial relation of occluded data.

VINCI '16, September 24 - 26, 2016, Dallas, TX, USA

ISBN 978-1-4503-4149-3/16/09…$15.00

DOI: http://dx.doi.org/10.1145/2968220.2968230

After reviewing related work in Section 2, we identify different causes of occlusion and introduce our novel visual-interactive design in Section 3. In Sections 4 and 5, we reflect on our approach and conclude with future work.

2. RELATED WORK

Occlusion is one of the main challenges in information-rich virtual environments, such as quantitative data embedded in 3D terrain [19]. Various approaches have been proposed to overcome its implications in related 3D visualizations. Elmqvist and Tsigas [6] present a comprehensive survey and taxonomy for occlusion management. They identify five distinct design patterns: tour planners, volumetric probes, multiple viewports, virtual X-ray tools, and projection distorters. Tour planners and volumetric probes are hardly applicable for our purpose, as they either require knowledge of all locations of interest in advance or affect the spatial reference of the embedded data. The last three patterns, on the other hand, are particularly related to our work and will be discussed in greater detail.

Multiple viewports provide several alternative views of a 3D environment simultaneously. Typically, one main viewport is used for first-person or detail views, whereas one or more secondary viewports show overviews of the environment. In virtual reality, common examples are worldlets [7] and worlds-in-miniature [22]. These techniques embed miniaturized 3D representations of specific objects or regions of interest in the main 3D visualization. Interacting with these representations allows users to inspect the environment from a vantage point and facilitates later recognition from an arbitrary viewpoint. The GeoZui3D system assists navigation and exploration of ocean data via frame-of-reference interaction [18]. Multiple windows can be created and linked to seamlessly change focus between different locations or to inspect hidden parts of an underwater environment. Navigating in a large virtual city model is addressed in [9]. An overhead map helps users in maintaining orientation while walking or driving through streets under limited visibility conditions. Such techniques are also powerful means for managing occlusion in terrain visualization, as terrain is essentially 2D in nature and lends itself to overhead maps. Yet, none of the existing approaches allows for identifying or highlighting occluded information via multiple views. Also, supporting 3D navigation through secondary views has only been sparsely addressed so far.

Virtual X-ray tools reveal hidden information by reducing the opacity of occluding objects. Generally, two types of approaches exist: *active* tools function as a user-controlled searchlight [5], and *passive* techniques automatically uncover invisible objects based on semantic information [8]. Especially active interaction allows for selectively removing occluding objects and thus, facilitates exploration. In [24], virtual X-ray is combined with stylization and smooth animation for emphasizing selected features in medical volume data. Managing occlusion in massive 3D point clouds is addressed in [4]. An interactive X-ray lens allows the user to inspect buildings or streets that would otherwise be covered by vegetation. Yet, applying virtual X-ray tools can have a negative impact on depth cues, which might impair depth perception and can hinder spatial relation. Moreover, using virtual X-ray techniques for quantitative data has not been thoroughly addressed so far. In this case, special care is required if multiple semitransparent objects are to be displayed. For instance, in terrain visualizations with high depth complexity, e.g., due to consecutive mountains, unintended mixing of colors might impair the interpretation of color-coded data.

Projection distorters integrate two or more different views of an environment into a single view, typically via nonlinear projection. The individual base views can often be actively selected by the user, which is particularly relevant in scenarios that focus on exploration. For example, multi-perspective 3D panorama maps allow for switching between degressive, progressive, and hybrid perspectives [15, 16]. Each choice supports a different navigation task, reduces occlusion, and improves utilization of screen space. Automatically determining the visibility of objects of interest, e.g., driving routes or hiking trails, in urban and rural environments is addressed in [2, 3]. Based on the computed results, local projection techniques are applied to generate views that ensure visibility and that introduce minimal distortion in the spatial surroundings. In [17] a multi-perspective overview+detail visualization is utilized for exploring 3D buildings. A secondary panoramic overview shows all facades of a building and facilitates navigation within a main 3D perspective view. In general, projection distorters are mostly used for discovery, rarely for access, and almost never for spatial relation [6]. Depending on the applied distortion, resulting visual displays can become confusing and disorienting.

Several other approaches for supporting exploration and navigation in 3D visualizations have been proposed. For instance, additional visual cues can be added for hinting at points of interest outside of the field of view [23]. Other examples include highlighting specific landmarks or modifying their appearance to aid orientation, e.g., by applying geometric scaling [10] or semantic levels-of-abstraction [21].

In summary, the three related design patterns offer different approaches for managing occlusion. Still, each solution covers only a certain aspect of occluded data in 3D terrain. Our goal is to develop an integrated approach that combines the individual strengths and thus, brings them to their full potential. To achieve this goal, we (i) introduce tailored concepts for capturing different types of occlusion, (ii) incorporate multiple occlusion management strategies, (iii) visualize occluded data via coordinated widgets, and (iv) augment each widget with appropriate interaction to ease navigation and to allow users to inspect details. With the resulting flexibility, we are able to tackle the peculiarities of quantitative geo-spatial data and to assist users in exploring 3D terrain visualizations.

3. VISIBILITY WIDGETS

We aim at supporting users in three main tasks for visually exploring virtual environments: *discovery*, *access*, and *spatial relation* of hidden information. To this end, we address three fundamental questions related to occlusion in 3D terrain: (i) how to detect occluded data, (ii) how to present occluded data, and (iii) how to interact with occluded data.

3.1 Detecting Occluded Data

Next, we describe different causes of occlusion and explain how occlusion can be detected on modern graphics hardware.

Causes of Occlusion. Different types of occlusion result from various scenarios while exploring 3D environments. Generally, we distinguish three causes: (i) occlusion by viewpoint, (ii) occlusion by terrain, and (iii) occlusion by embedded data.

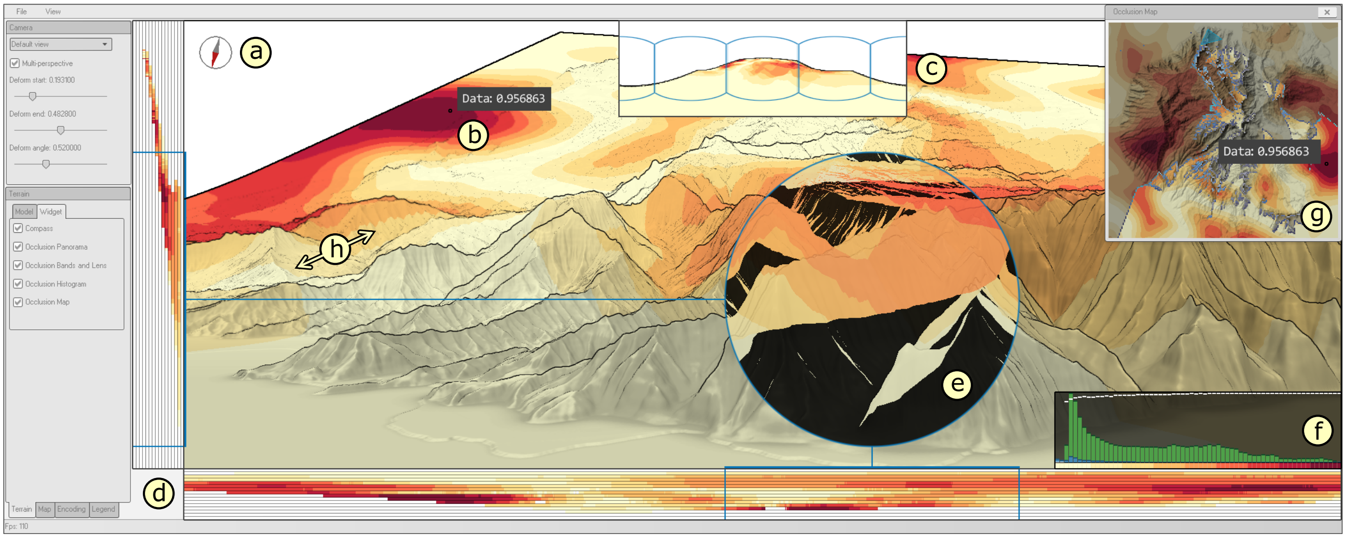

Figure 1: Schematic overview of our visualization tool. The main 3D view shows color-coded data embedded in terrain (middle). The user interface supports loading different terrain models, adjusting the visual encoding, and switching between multiple data attributes (left). An interface skin with light colors helps to focus on the main visualization. Our tool integrates four widgets and a rich set of coordinated interactions for showing hidden information and details on-demand: (a) compass, (b) tooltips, (c) panorama view, (d) bands view, (e) interactive lens, (f) histogram view, (g) 2D overview map, and (h) multi-perspective 3D view.

Occlusion by viewpoint emerges from seeing only parts of the data with a chosen camera configuration. Digital terrain models are typically large and can not be displayed in their entirety due to limitations of available display space. Choosing different viewpoints, i.e., viewing positions and orientations, is therefore necessary either to review the encoded data in overviews or to inspect details in close-up views. Yet, no matter what viewpoint is chosen, none will show all data values at once and thus, occlusion prevails. In this regard, we consider the spatial context of the data. That is, we distinguish between: (i) data located outside of the field of view and (ii) data located inside the field of view. Data outside of the field of view is always invisible, as it is not covered by common rendering methods. Data inside the field of view may end up visible or invisible in the rendered image, depending on the other two causes of occlusion. For instance, viewpoints with locations near the terrain's surface in combination with orientations towards the horizon are particularly problematic. In contrast, top-down perspectives rarely lead to high occlusion ratios. Still, in both cases important information might get missed without adequate detection mechanisms.

Occlusion by terrain results in the inability to see data values due to occluding parts of the terrain. Dealing with the resulting implications is the primary focus of our work. Interestingly, from a spatial interaction point of view, we are dealing with a singleton environment containing solely the 3D terrain. Thus, only self-occlusion can occur depending on the placement of individual geometric features, e.g., mountains and valleys, from some viewpoint. Other types of spatial interactions, such as intersection, enclosement, and containment (cf. [6]), have secondary role in our setup. The main influencing factor is the geometric complexity of the terrain. With geometric complexity, we refer to the general surface shape as well as to the detail level of the terrain's local relief. The more complex a terrain is, the more information can it potentially encode, and the larger is the impact of occlusion. For coarse-grained terrain with low elevation, occlusion of mapped data mostly occurs due to back-facing surface parts. In contrast, the finer the geometric features and the steeper the slope, the greater is the chance of self-occluding front-facing surface parts, and hence, the more extensive will the data loss be.

Occlusion by data follows from multiple data values competing for the encoding of a single pixel on screen. Such situations can arise either due to: (i) data values from disparate attributes, (ii) combinations of multiple data values, or (iii) perspective projection. In the first case, values from more than one attribute in a multivariate data set cover the same spatial region, but can hardly be combined. In the second case, values are combined, but extracting individual values might no longer be possible. For example, certain occlusion management strategies can reveal multiple occluded data values at once, typically by increasing the transparency of occluders. This might lead to revealed data values overlapping with each other or with data that is already visible. While this can be a valuable approach for discrete objects, mixing colors or texture of quantitative data is often disadvantageous. In the third case, over-plotting particularly impacts data values that are distant to the viewer and thus, have to be presented in a small area of the screen. To avoid adverse occlusion-related effects in all three situations, special attention and care is required. Otherwise, common rendering pipelines may produce unexpected results or follow unwanted default behavior, such as displaying averaged or nearest samples of the data. As a result, information of interest might get obscured by other irrelevant data.

In sum, we are covering proximity-related occlusion inside and outside of the field of view, resulting from the geometric complexity of the terrain and the density of the data.

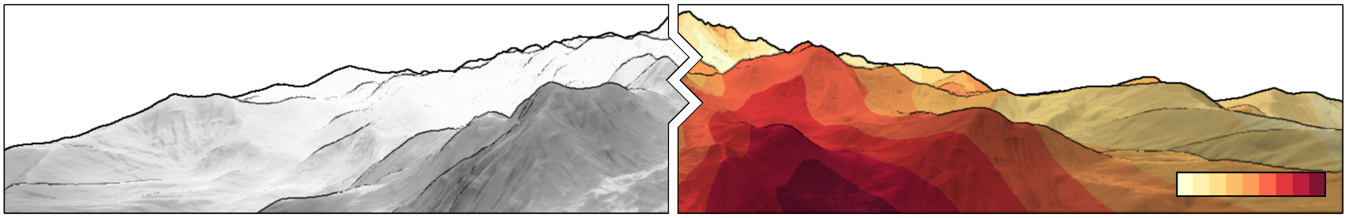

Figure 2: Visualizing terrain and data. The geometry of the terrain is encoded via directional lighting, ambient occlusion, and enhanced edges (left). The data are directly mapped onto the terrain's surface via color-coding (right), with low values shown in light yellow and high values in dark red. Stylizing the terrain reduces illumination details with distance to the viewer for approximating aerial perspective and facilitating interpretation of the color-coded data.

Computational Occlusion Detection. We apply a general pipeline-based computation for detecting the different types of occlusion. Following the basic model for 3D occlusion in [6], we consider data of the following form. A terrain T is represented by its surface in Cartesian space $(x, y, z) \in \mathbb{R}^3$. A data set $D = \{d_1, \ldots, d_n\}$ consists of n data points. Each data point d_k with $1 \leq k \leq n$ stores at least one data value and a location on the surface of T. A viewpoint $v = (M, P)$ is given by a view matrix M and a projection matrix P. Both matrices describe the position and orientation of the viewing frustum v_f of a virtual camera. A data point d_k is classified as *visible* from a viewpoint v if there exists a line segment between v and d_k that does not intersect any part of T, and *occluded* if no such line segment exists. In addition, d_k is considered *inside the field of view* if its location is contained in v_f, and *outside the field of view* otherwise.

To detect occluded data, we first generate a reduced data set D_r by filtering out all values of attributes that are not under immediate consideration. We then project D_r into image space according to M and P of v. As a result, we get a set C of potentially contributing data points for a target image I. According to the boundaries of I, we subdivide C into C_{in} and C_{out}, containing data points inside or outside of the image, respectively. We further sort C_{in} by distance to v and classify the data points into visible C_{in}^v or occluded C_{in}^o. Altogether, we obtain four subsets: for inside C_{in} and outside C_{out} in general, and for visible C_{in}^v or occluded C_{in}^o within the image. To determine which data points are to be considered in each view, we forward the four subsets to the different occlusion management strategies instantiated by our widgets (see Sect. 3.2). In return, we get a set of data points $C_{p_{ij}}$ for every pixel p_{ij} in $I = \{p_{11}, \ldots, p_{wh}\}$ with $1 \leq i \leq w$ and $1 \leq j \leq h$. If $C_{p_{ij}}$ contains only a single data value, it is directly encoded in the respective view. In case multiple data values remain, i.e., occlusion by data occurs, additional user-specified filter mechanisms may be applied, e.g., maximum filter.

We efficiently compute the subsets via a combination of customized rendering techniques. For multivariate data sets, we encode all data values per considered attribute in individual textures. This allows us to switch between them instantaneously. If possible, we reduce overall computational costs by performing calculations in image space, following a deferred rendering strategy. Data points inside (C_{in}) and outside (C_{out}) of the field of view are processed via dynamic environment mapping. To determine C_{in}^v and C_{in}^o for each pixel in I, we consider multi-pass approaches, e.g., depth peeling, as well as single-pass techniques, e.g, layered rendering. Generating specialized mipmaps and integrating respective sampling techniques allows us to realize different types of filters for managing occlusion by data, i.e., in case of $C_{p_{ij}} = \{a_1, \ldots, a_m\}$ with $m > 1$. If required, more sophisticated aggregations, e.g., histograms, and summary statistics can be computed in parallel on the graphics hardware (e.g., [20]). Calculated occlusion from different viewpoints, e.g., 3D views and 2D views, is transferred via customized shadow mapping. The synergy of these diverse techniques allows us to perform occlusion detection in real-time. That is, if necessary, all calculations are executed each frame to facilitate interactive exploration.

3.2 Presenting Occluded Data

We design four coordinated visibility widgets to communicate the detected occlusion of data: a panorama, horizontal and vertical bands, a histogram, and a map. Each widget incorporates a different occlusion management strategy and consists of a dedicated auxiliary view, which can be activated on-demand. Figure 1 shows an overview of our visualization tool for quantitative data embedded in 3D terrain. Next, we describe the basic terrain visualization and the visibility widgets in detail.

3.2.1 Basic Encoding of Terrain and Data

Encoding Terrain. We apply various shading techniques and edge enhancement for visualizing terrain. Our primary shading aims at illustrating surface properties. For this purpose, we consider two illumination methods: a simple local technique based on a directional lighting model and a more sophisticated global approach based on ambient occlusion. Directional lighting depicts the general shape of the terrain's surface, whereas ambient occlusion highlights details and relationships between neighboring parts. Additional stylization allows us to further emphasize certain terrain properties. On the one hand, we utilize a customized toon shading technique to create various view-dependent effects, including levels-of-abstraction, aerial perspective, or depth-of-field [1]. For instance, modifying the influence of illumination depending on the distance to the viewpoint results in a continuous abstraction that also reinforces perception of depth. On the other hand, enhancing edges makes it easier to distinguish individual parts. For example, different edge styles allow for tracing ridges or providing additional shape detail. The stylizations can be interactively adjusted or completely deactivated. The user can switch between effects, set the amount of abstraction being applied, and regulate which edges are

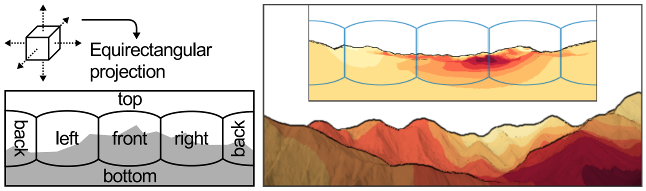

Figure 3: Visibility panorama. The diagram illustrates the generation of the 360° overview via environment mapping and equirectangular projection (left). Activating the widget allows for inspecting relevant information in the immediate vicinity of the viewer (right).

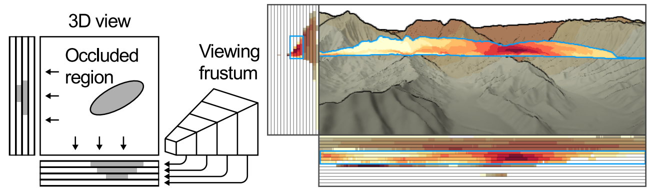

Figure 4: Visibility bands. The diagram illustrates how occluded information inside the camera's viewing frustum is successively projected onto both image axes (left). The widget visualizes the projected information via horizontal and vertical bands, each with 16 layers ordered by distance to the viewer (right). When selecting one or multiple layers related occluded data are shown in the 3D view.

to be shown. Combining the different encoding techniques allows for effectively communicating inherent geometric features of terrain. Figure 2 shows an example of our visual design for terrain and data.

Encoding Data. We select color for mapping quantitative data directly onto the terrain's surface. For encoding visible values (C^v_{in}), we apply color palettes from ColorBrewer [11]. Data values are assigned to a limited set of colors per palette to support their identification in an associated legend (Fig. 2). In case of multivariate data, attributes are shown sequentially and are allocated to unique palettes. By stylizing the terrain, the communication of data is further facilitated. For example, attenuating distant shading allows for interpreting faraway data more easily, while providing nearby surface details enables relating data and geometric details. Moreover, depicting the terrain solely via gray scales ensures sufficient visual contrast to the color-based encoding of the data. If required, the default color-coding can be interactively adjusted or matched with color conventions in the problem domain.

3.2.2 Design of Visibility Widgets

Visibility Panorama. To manage occlusion by viewpoint, we integrate a multi-perspective panorama as a widget into the main 3D visualization. Generating the panorama is based on a customized environment mapping technique. With regard to the main virtual camera, we render the scene in all six viewing direction, i.e., front, back, left, right, up, and down. All rendered images are then joined into a panoramic map via equirectangular projection. This projection method allows for a compact depiction with only low horizontal and varying vertical distortion. To support the interpretation of the panorama, borders of each viewing direction can be superimposed. Figure 3 illustrates the resulting display. Activating this widget provides a 360° overview of the terrain. Information outside the main view can be identified without changing the camera's orientation. At the same time, information inside the main view can be related to the surroundings. Focusing on the data is facilitated by abstracting from terrain details in the panorama, e.g., by showing only main slope lines. Moreover, the widget can be set to show only visible, only occluded, or all information regardless of occlusion. We employ rendering filters for showing all information or highlighting selected ranges, which, however, come at the cost of reduced depth perception (see Sect. 3.1). This way, direct access to visible as well as occluded information located in the immediate vicinity of the user is granted.

Visibility Bands. To address occlusion by terrain, we introduce horizontal and vertical visibility bands with an indirect reference to space. First, we determine all occluded data values inside the viewing frustum of the camera. These values are then projected into 2D image space and finally horizontally and vertically onto two 1D value sets. Both value sets are visualized as visibility bands that depict the distribution of occluded information for each image axis. Aligning the bands along the border of the main view, allows users to overview the encoded values and to relate them to the rendered image of the 3D scene. To give the user a better idea at what distance occluded data are located, we additionally subdivide the viewing frustum into equal depth ranges. The occluded values in each depth range are assigned to respective layers in each band. The layer with the closest depth range is placed next to the image border and the remaining layers are arranged by distance to the viewpoint. The number of layers and the total covered depth range can be set by the user. This way, the 3D location can be approximated, with only a marginal increase in required display space. Figure 4 shows our design of visibility bands.

Visibility Histogram. With the histograms we primarily target occlusion by data. The fundamental idea is to abstract from space and take a more data-oriented approach for analyzing hidden information. We achieve this by setting values that are visible in the main view in relation to all values located in the camera's viewing frustum. Binning these values allows us to focus on the data distribution. In addition, we compute the ratio of the visible parts and the total amount of values for each bin. These ratios represent the percentage of occlusion for the respective value intervals. The number of bins can be adjusted by the user. Figure 5 shows the computation of the histograms and their combined visualization with the ratios as a widget. The greatest advantage of this widget is the compact representation of all information in the current viewing direction. Looking at the histograms allows users to identify, if all values of interest are actually visible in a chosen view of the scene. In case a specific value interval exhibits occlusion beyond a tolerable threshold, more suitable views can be determined in concert with the other widgets. Utilizing the visibility histograms this way allows for verifying hypotheses and compensates for the inherent abstraction of space.

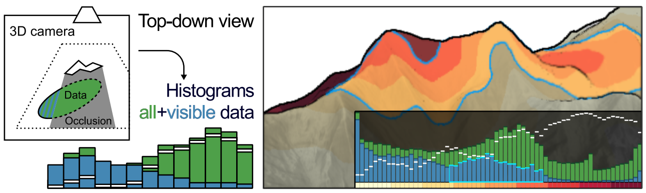

Figure 5: Visibility histogram. The diagram summarizes the computation of the histograms (left). The widget shows the relationship between distributions of visible values (blue) and all values (green) inside the current view (right). A white marker per bin indicates the percentage of occlusion in the respective value intervals. A narrow color legend helps associating bins and data values (bottom). When selecting one or multiple bins the respective data values are highlighted in the 3D view.

Visibility Map. Integrating a secondary 2D map view can be an effective support for exploring 3D terrain visualizations. The map shows the terrain from a top-down perspective via an orthogonal projection. This generates an overview of terrain and data, aids navigation, and thus handles both: occlusion by viewpoint as well as occlusion by terrain. Illumination of terrain and encoding of data are applied similar to the main 3D view. Yet, the direction of the lighting is fixed to a north-west incidence angle by default. This design follows general shading guidelines for cartographic maps and helps to differentiate craters and mountains [12]. Furthermore, the coloring can be switched to encode elevation of the terrain. In this case, an alternative encoding for data is applied, e.g., isolines. To further increase the maps utility, we augment the visualization with spatial occlusion information from the main view. The hidden parts of the 3D scene are determined and transferred via a customized shadow mapping technique. Activating this functionality, desaturates and darkens respective parts of the map. Additionally, a miniaturized viewing frustum depicts the location and orientation of the main camera. Figure 6 shows the resulting design as a widget.

Individually, the four auxiliary visualizations implement powerful occlusion management strategies. Their real benefit comes into effect when being applied in concert. For example, the map does not directly allow users to analyze occluded information with respect to the terrain's geometry. But, activating the panorama or bands supports locating that information in the main 3D view. As explained next, coordinated interaction further assists in relating encoded information in each widget and in the main 3D visualization.

3.3 Interacting with Occluded Data

To support a comprehensive visual exploration, we provide a rich set of tailored interaction techniques. While we focus on communicating occlusion with our widgets, coordinated interaction is essential for discovery, access, and spatial relation. Besides widget-related modifications, e.g., activating and resizing, we particularly support navigation as well as selecting and highlighting to inspect details on-demand.

Navigation. In the main 3D view, the virtual camera can be interactively controlled to freely move through the 3D terrain and to take different points of view on the embedded data. Additionally, predefined camera settings can be applied, e.g., bird's eye view or landmark views. Every movement of the camera is smoothly animated in order to prevent sudden changes in the rendered image. A compass can be superimposed as an additional visual cue for maintaining orientation (Fig. 1a). In the 2D map view, we support navigation either independent of or dependent on the main camera. The first option consists of interactive zooming and panning to adjust the visible section of the terrain. For the second option, the map is coupled with the 3D view to keep it centered at the current position of the main camera. All other visualizations are linked with the main 3D view and automatically update according to user input.

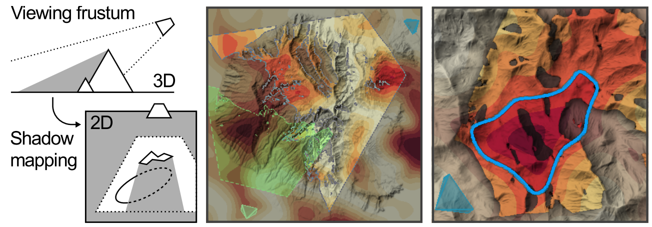

Figure 6: Visibility map. The diagram illustrates how occlusion in the main 3D view is transferred to the 2D map view via customized shadow mapping (left). The visibility information is visualized by highlighting parts in sight and by desaturating and darkening parts that are obscured or outside of the field of view (middle). Miniaturized viewing frustums indicate the current viewpoint of the main 3D camera (blue), and an interactively specified destination (green). When selecting spatial regions respective parts of the map are outlined, which allows users to check their visibility (right).

The panorama widget and the map widget provide two additional features for aiding navigation through the 3D terrain. First, clicking on a point of interest in the panorama centers the 3D view at the selected location via a continuous rotation of the main camera. This particularly eases inspecting information outside of the field of view, e.g., next to or below the viewer, without having to search for suitable viewpoints manually. Second, the map allows users to interactively specify a destination, i.e., a target position and orientation, for the main 3D camera. Activating this feature shows a second miniaturized viewing frustum and highlights all parts of the terrain that are in sight at the currently selected destination (Fig. 6). This ensures that the selected parts are actually visible once arrived. Confirming the destination automatically moves the main camera along a collision-free path. This way, even remote locations can be conveniently explored without having to navigate through the 3D terrain by hand and repeatedly checking the map if the correct viewpoint has already been reached.

Altogether, our navigational aids support discovering occluded data and thus, are particularly helpful in managing occlusion by viewpoint and by terrain.

Selecting and Highlighting. We enhance the visualization of the interlinked widgets with flexible selection and highlighting facilities. We support selecting individual points in all views and show their assigned values via tooltips. To help associating selected points of the 3D view with the 2D map view, we transform the current selection from one view to the other and duplicate the cursor plus tooltip at the respective location (Fig 1b).

Besides individual points also spatial regions can be selected. In the main view and the map view, we support this directly by: (i) polygon selection and (ii) value range selection. The polygon selection allows users to interactively set multiple corners in the terrain to define a spatial region of interest. The value range selection requires only a single seed point as input and automatically forms the respective region by connecting all neighboring parts within a certain threshold. Selecting spatial regions this way, highlights related parts in the 3D terrain and in the 2D map (Fig. 6). The utility of these features is twofold: (i) selections of the map can be related to the 3D geometry of the terrain and (ii) selections of the 3D terrain can be inspected with regard to their surroundings in the map.

Inspecting specific image regions is provided by an *interactive X-ray lens*. Activating the lens in the main view allows users to investigate hidden information locally by superimposing and highlighting occluded parts in the rendered image. The position and size of the lens can be adjusted. If the bands widget is active, the relationship between the bands and lens is illustrated by connecting lines (Fig. 1e). Optionally, the lens can be configured to show occlusion of one layer, of multiple layers, or aggregated across all layers of the bands. This permits users to examine detected patterns in the bands as well as in the rendered images.

Further specialized approaches for inspecting hidden information in particular are instantiated by the bands widget and the histogram widget. The bands allow for selecting one or multiple layers to highlight the associated parts in the terrain. This helps users to judge which regions are covered by the layers. Picking individual sections of the bands is enabled either by clicking on colored segments or by using a selection rectangle. Starting this selection in the horizontal band automatically determines and highlights all associated sections in the vertical band and vice versa for clarifying their correspondence. Figure 4 illustrates selecting and highlighting occluded information via the bands widget.

The histogram widget allows for selecting single or multiple bins either from the visible distribution, the occluded distribution, or both distributions in combination. All data values that contribute to the chosen bins are highlighted in the terrain. If visible as well as occluded bins are selected, a user-specified filter, e.g., a maximum filter, is applied to decide which value to encode in each pixel of the display (see Sect. 3.1). In addition, a modified color-coding ensures that occluded parts can be distinguished from visible parts. Figure 5 shows this functionality.

All selections can be expanded or reduced using binary operations. While selecting, the 3D view and the 2D map view are updated to outline selected parts and dim non-selected parts. Essentially, this functionality allows users to relate the abstracted information encoded in the widgets back to space and to reconstruct its spatial context to a certain extent. In sum, our selection and highlighting techniques enable users to access and associate data shown in the interlinked views as well as to spatially relate them with the geometry of the terrain. Hence, these features particularly address the inspection of values and occlusion by data.

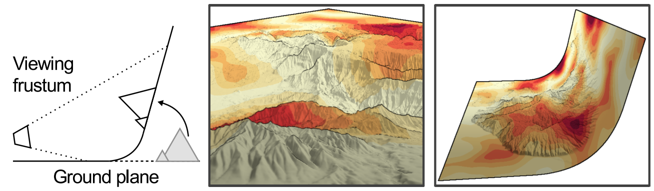

Figure 7: Additional occlusion management strategies. The diagram outlines the generation of a 3D panorama by deforming the terrain geometry with regard to a parametric curve [16] (left). The resulting visualization creates a focus zone, a transition zone, and a context zone with increasing distance to the viewer (middle). To exemplify the deformation the reference point is set to a fixed location, independent of the virtual camera (right).

Augmenting our detection and presentation of occluded data with this rich set of interaction techniques completes our visualization tool. Next, we reflect on the utility of our visibility widgets for exploring data embedded in 3D terrain and discuss future work.

4. DISCUSSION

We developed our tool in the context of collaborative work with experts from the aviation industry. Following a participatory design, we jointly specified suitable strategies for occlusion management and devised corresponding visualizations. During this process, we built upon their domain expertise, identified challenges, and gathered informal feedback. We ascertained that exploring data embedded in 3D terrain is an important task in various aviation-related scenarios, e.g., flight route planning. Particularly during take-off and landing, the analysis of weather data plays a vital role for determining suitable routes and courses of action. In such situations, it is essential that relevant information, e.g., hazardous phenomena, does not go unnoticed. Hence, discovering, accessing, and relating occluded data needs to be supported. The abilities to address these tasks with our tool, to interactively visualize occluded data even in extensive terrains, and to reduce the need for 3D navigation in general were identified by our collaborators to be major advantages.

While we focused on managing occlusion of data embedded in 3D terrain, our solutions can be adopted for handling other setups as well. For example, the spatial frame of reference may be directly changed to virtual city models or underwater landscapes. The geometry representing such environments can be comparatively more complex. In this regard, our detection and visualization are applicable to occlusion by multiple objects, e.g., added obstacles or landmarks. Objects with no intrinsic information value can be dynamically flagged as occluders and treated in a similar manner as the terrain for the computation of occluded data (see Sect. 3.1). In addition, our auxiliary widget design allows for combinations with other occlusion management strategies that alter the main 3D view directly. Figure 7 illustrates 3D panorama maps [16] and Figure 1 shows them incorporated in our visualization tool. If both strategies are applied, occluded data can be discovered via our widgets and directly accessed via the 3D panorama map.

5. CONCLUSION AND FUTURE WORK

We presented novel visual-interactive support for managing occlusion in 3D terrain visualization. The combination of different strategies opens up new possibilities for detecting, presenting, and interacting with occluded data. Our systematic detection and efficient computation enables processing various types of occlusion by viewpoint, by terrain, and by data in real-time. With the help of four complementary widgets, users can overview visible and occluded data inside and outside of the field of view, and locate and analyze that data in the main 3D visualization. Coordinated interaction facilitates selecting and highlighting in the interlinked views, investigating details on-demand, and eases navigation. Integrating the widgets in our flexible visualization tool allows for a full-fledged management of *all* types of occlusion of quantitative data in 3D terrain, for which to the best of our knowledge no equivalent approach exists. We conclude that our approach can be a useful aid for exploring 3D terrain.

A limitation of our approach is that only a single data attribute can be inspected at a time. More work is required for supporting multiple data attributes, qualitative data, and associated uncertainty simultaneously. Visualizing occlusion of multivariate data could be addressed via specific encoding, e.g., bivariate color-coding, or additional views, e.g., one pair of bands for each attribute in the bands widget. Detecting occlusion of qualitative data is possible by computing the contribution of the individual values to each pixel on screen and integrating specialized filters to deal with their overlap.

To further improve our approach, we plan to incorporate automated cues for hinting at occluded data. For example, while exploring 3D terrain with our tool, simplified visual indicators can be shown to make users aware of hidden information. Clicking on an indicator will suggest an appropriate widget based on the given situation. This allows users to initially check and analyze the data, and if required, retrieve further details on-demand. Moreover, it will help users to focus on relevant information by showing only those additional views that are currently needed. In this context, studies for evaluating which occlusion management strategy matches which situation and analysis task will become necessary.

6. REFERENCES

[1] P. Barla, J. Thollot, and L. Markosian. X-toon: An extended toon shader. In *Proc. of ACM NPAR*, pages 127–132, 2006.

[2] H. Deng, L. Zhang, C. Han, Y. Ren, L. Zhang, and J. Li. Efficient occlusion-free visualization for navigation in mountainous areas. *Comput. Geosci.*, 52:389–397, 2013.

[3] H. Deng, L. Zhang, X. Mao, and H. Qu. Interactive urban context-aware visualization via multiple disocclusion operators. *IEEE Trans. Visual Comput. Graphics*, PP(99):1–1, 2015.

[4] S. Discher, R. Richter, and J. Döllner. Interactive and view-dependent see-through lenses for massive 3D point clouds. In *Proc. of JIGC*, 2015.

[5] N. Elmqvist, U. Assarsson, and P. Tsigas. Employing dynamic transparency for 3D occlusion management: Design issues and evaluation. In *Proc. of IFIP INTERACT*, pages 532–545, 2007.

[6] N. Elmqvist and P. Tsigas. A taxonomy of 3D occlusion management for visualization. *IEEE Trans. Visual Comput. Graphics*, 14(5):1095–1109, 2008.

[7] T. T. Elvins, D. R. Nadeau, and D. Kirsh. Worldlets: 3D thumbnails for wayfinding in virtual environments. In *Proc. of ACM UIST*, pages 21–30, 1997.

[8] S. K. Feiner and D. D. Seligmann. Cutaways and ghosting: satisfying visibility constraints in dynamic 3D illustrations. *Vis Comput*, 8(5):292–302, 1992.

[9] S. Fukatsu, Y. Kitamura, T. Masaki, and F. Kishino. Intuitive control of "bird's eye" overview images for navigation in an enormous virtual environment. In *Proc. of ACM VRST*, pages 67–76, 1998.

[10] T. Glander, M. Trapp, and J. Döllner. A concept of effective landmark depiction in geovirtual 3D environments by view-dependent deformation. In *Proc. of Int. Symp. on LBS and Telecartography*, 2007.

[11] M. Harrower and C. A. Brewer. Colorbrewer.org: An online tool for selecting colour schemes for maps. *Cartogr J*, 40(1):27–37, 2003.

[12] E. Imhof. *Cartographic Relief Presentation*. ESRI Press, 2007.

[13] M. Kreuseler. Visualization of geographically related multidimensional data in virtual 3D scenes. *Comput. Geosci.*, 26(1):101–108, 2000.

[14] H. Lorenz and J. Döllner. 3D feature surface properties and their application in geovisualization. *Comput. Environ. Urban Syst.*, 34(6):476–483, 2010.

[15] S. Möser, P. Degener, R. Wahl, and R. Klein. Context aware terrain visualization for wayfinding and navigation. *Comput. Graph. Forum*, 27(7):1853–1860, 2008.

[16] S. Pasewaldt, A. Semmo, M. Trapp, and J. Döllner. Multi-perspective 3D panoramas. *Int J Geogr Inf Sci*, 28(10):2030–2051, 2014.

[17] S. Pasewaldt, M. Trapp, and J. Döllner. Multi-perspective detail+overview visualization for 3D building exploration. In *TPCG*, 2013.

[18] M. Plumlee and C. Ware. Integrating multiple 3D views through frame-of-reference interaction. In *Proc. of CMV*, pages 34–43, 2003.

[19] N. F. Polys and D. A. Bowman. Design and display of enhancing information in desktop information-rich virtual environments: Challenges and techniques. *Virtual Real*, 8(1):41–54, 2004.

[20] T. Scheuermann and J. Hensley. Efficient histogram generation using scattering on GPUs. In *Proc. of I3D*, pages 33–37, 2007.

[21] A. Semmo, M. Trapp, J. E. Kyprianidis, and J. Döllner. Interactive visualization of generalized virtual 3D city models using level-of-abstraction transitions. *Comput. Graph. Forum*, 31(3):885–894, 2012.

[22] R. Stoakley, M. J. Conway, and R. Pausch. Virtual reality on a WIM: Interactive worlds in miniature. In *Proc. of CH*, pages 265–272, 1995.

[23] M. Trapp, L. Schneider, C. Lehmann, N. Holz, and J. Döllner. Strategies for visualising 3D points-of-interest on mobile devices. *J Location Based Services*, 5(2):79–99, 2011.

[24] I. Viola, M. Feixas, M. Sbert, and M. E. Gröller. Importance-driven focus of attention. *IEEE Trans. Visual Comput. Graphics*, 12(5):933–940, 2006.