A Self-adaptive Technique to Visualize Geospatial Data in 3D With Minimum Occlusion

Abon Chaudhuri & Han-wei Shen
The Ohio State University
Tech Report OSU-CISRC-1/10-TR02

Abstract

Geospatial data, a popular information structure in recent times, is commonly rendered as a color-coded 2D map. The secondary information available at a location, for example - the temporal variation of a spatial attribute, is displayed on-demand. When a few such displays are present, occlusion of the 2D map by them can seriously affect further exploration. Even though the degree of occlusion varies with the type of the display, the more informative and useful ones, such as rectangular windows consume more screen space and cause higher occlusion. An occluded portion can be revealed by moving the display away from its originating location, however that increases the cognitive load of visually linking the displayed information to the location of interest. This paper presents a self-adaptive technique that minimizes occlusion by automatically distorting the spatial map to a reasonable extent so that a usable screen space is created for the requested visualization. Unlike existing methods, the created space stays adjacent to the location of interest and hence, enables the user to easily relate the secondary display to the spatial location. Furthermore, the proposed method empowers the user to control the existing trade-off between occlusion and spatial distortion. The paper demonstrates two potential applications, one of which includes a novel mirror-like interface designed for comparative analysis of spatio-temporal data.

1 Introduction

Geospatial data analysis and visualization have gained as much pertinence in recent years as never before. While visualization, data mining and GIS researchers have proposed novel techniques for analyzing and presenting geospatial data in various forms, numerous softwares like Google Map and Earth [Inc09], Microsoft Bing Map [Cor09], ArcGIS [ESR09], NASA Worldwind [NAS09] have attempted to enhance the scope for the developers and the common users.

Even though geospatial data can be displayed in different ways depending on the objective, the popularly used style is to present it as a 2D cartographic map (Figure 1(a), a screenshot from Google Map) or to embed it on a 3D sphere to be explored from a bird's eye view (Figure 1(b) from Google Earth). Depending on the available information, a suitable attribute such as temperature, air pressure at each spatial location of the map can be used to create either a color or a height map. But in most cases, encoding an attribute to color or height is not enough since the information to be presented at a spatial location can be far more complex. For example, temporal variation of an attribute (or even multiple attributes) at each spatial location often comes as an integral part of the data.

Different approaches have been tried with the objective of creating extra room for displaying such secondary information. Multiple coordinated display is one technique where the user can select a spatial location on the map to find the corresponding information highlighted on other windows. This method can lead to a larger cognitive burden, especially when multiple locations have been selected since keeping track of the correspondence may become difficult for the user.

Another popular approach is to pop up a new window each time the user clicks on a location. The cognitive burden of linking a location to a point in a remote window is reduced by this method, since the spatial location is now tied to the window by a visible link. But Figures 1(a) and 1(c) suggest that this method introduces another serious problem, that of occlusion. One window can potentially occlude a portion of the spatial map and also, another window. Popular softwares employ one or more of the following strategies to deal with occlusion:

1. They allow a viewer to fly around a location of interest and look at it from all directions to re-

veal any potentially occluded zone.

2. They use small suggestive icons which change to larger windows on click. However some regions of the map can get cluttered due to many such icons (the circled region in Figure 1(a)) making the selection of a single one difficult. Moreover, such icons themselves are small and hence, very easy to get occluded behind larger windows (see Figure 1(c)).

3. They limit the number of informative windows that can be displayed at a time. For example, the default setting of Google Earth automatically closes the previous window when a new one appears. This works fine for casual browsers; but for someone who is exploring the map with an analytical goal may require the facilities of comparison and multiple foci.

4. Some softwares have introduced semi-transparent windows, which can be overlaid on top of the 2D map. This method has two limitations. First, all sorts of visualizations cannot be overlaid. For example, if a user requests a snapshot of a region of the map from a different time step, the fetched map should not be overlaid on the current map. Second, as more and more semi-transparent layers are placed on top of the map, the original map no longer remains clearly visible through them.

Another alternative proposed by the researchers is to replace the rectangular windows by tall and thin shaped icons [TSWS05], which would not occlude each other and remain visible because of their height. Again, not all types of information can be suitably represented using such non-standard structures. Moreover, the user needs to familiarize himself with the structure each time he encounters a new type.

The discussion so far indicates that a number of ad-hoc techniques have been developed and commercially released instead of one general method that can deal with occlusion. This paper introduces a technique which significantly reduces occlusion by allowing the spatial map to adapt itself to user interaction. In the proposed method, as the user clicks on a particular location or a region to visualize the secondary information in 3D, the 2D map distorts within a reasonable limit to accommodate the appearing visualization. The benefits of the method include:

1. The appearing window does not occlude the portion behind it. The technique is view-dependent since the spatial map adapts to maintain its visibility as the user changes the viewpoint.

2. The appearing window always stands adjacent to its originating location. Hence, the user is spared of the task of visually linking two remote points.

3. The pop-up window can be utilized in different ways. The paper demonstrates two examples. The first one is a novel approach which uses the windows as mirror-like interface for better comparative analysis of spatio-temporal data. The second one uses the technique to reduce visual clutter.

The benefits come at the cost of distortion of the spatial map though. Our technique also empowers the user with different parameters, which can be tuned to find an optimal point between distortion and occlusion.

The basic intuition is borrowed from the work by Chaudhuri and Shen [CS09] in which the deeper levels of a treemap are displayed on-demand in 3D and the treemap is partially elevated to avoid occlusion. Unlike treemaps, spatial maps contain geographic features with no rigid boundary. Hence, abrupt changes of elevation are not acceptable in this case. Moreover, the viewer of a spatial map must be allowed to navigate freely in 3D space unlike the unidirectional viewer of treemaps. To claim its originality, this paper deviates from the mentioned work to meet these requirements.

The rest of the paper is organized as follows: Section 2 summarizes the research progress in geospatial visualization in recent years. Section 3 introduces the technique. The proposed method, which is a blend of algorithm and visual design, required us to carefully take a few subjective choices, discussed in Section 4. Section 5 demonstrates two possible applications of our method, followed by a conclusive discussion.

2 Related Work

As geospatial data has grown in size and diversity over decades, the traditional cartographic 2D drawing of maps has been improved with the aid of interactive graphics and visualization technology [Mac01]. One class of techniques has enhanced the representation of the 2D map, while the other has proposed new methods for visualization of the secondary information, which is often as complex as time series, multivariate or network data, embedded on the spatial map.

The 2D techniques often use non-standard methods of color [Bre94] [SRd05] [DB07] or height mapping to focus on different aspects of the data. Dis-

(a) (b) (c)

Figure 1: Illustration of the problem of occlusion. **(a)** 2D cartographic representation of Amazon Basin from Google Map. The selected image occludes the map and other icons. **(b)** 3D representation of a town on Amazon from Google Earth. The lower left region contains numerous icons. **(c)** The problem of occlusion affects 3D maps also, when a window appears.

torting the geographic map by encoding a spatial attribute to area or distance leads to Cartogram, which has been thoroughly explored in Worldmapper [DBN06]. Panse et. al. [PSKN06] have proposed pixel-based techniques for spatial distortion of geographic map. Besides the idea of distorting the entire map, several focus+context methods which introduce local distortion of the focus area have been adapted to geovisualization. A recent work [QWC+09] has utilized spatial distortion to answer user-queries on finding the best route between two geographic locations.

When extra screen space is consumed to represent the secondary information, it is either done in the form of window(s) appearing at the selected spatial location (as in Google Map and Google Earth), or as separate window(s) in co-ordination with the spatial map(as in GeoVista Studio [GTWH02]). Legible Cities [CWK+07] is an example which exploits multiple co-ordinated display for presenting spatially embedded multi-dimensional data. Spatio-temporal data have been explored with multiple co-ordinated displays in [AA05] [SFdOL04].

Occlusion is as serious a concern to 3D geovisualization as it is to other 3D interaction techniques. A comprehensive survey of the approaches to deal with occlusion can be found at [ET08]. Intuitive metrics of measuring occlusion [ED06] [Bra97] have been proposed to enable comparison of traditional 2D techniques. Like many other techniques, jittering of points [TGC03] is not directly applicable to geographic data, where the occlusion is caused by standing icons or windows. Pencil-shaped or helical icons [TSWS05], pin-shaped probes [BDW+08] and connected line segments [KW04] have been introduced because such structures cause comparatively less occlusion. An occlusion-culling technique [PGSF04] has been proposed for geoscientific data with sub-surface information layers.

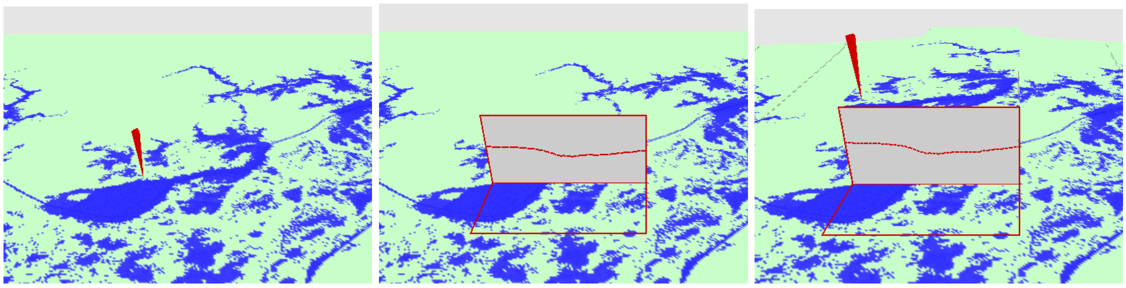
3 Proposed Method

In our technique, the geometry and orientation of the primitives constituting the spatial map automatically adjust to reveal the occluded portions, when a 3D window partially obstructs it. Figure 2(a) displays a portion of a spatial map of Amazon River Basin with a red landmark placed by the user. The spatial map is constructed from a 460×900 grid which contains the water elevation data of Amazon basin for a single day. The dataset contains 486 temporal instances of the spatial grid. In essence, each vertex of the grid associates a time series of daily water elevation with it. Naturally the user would be interested in studying the time-varying data, which can be presented in a window that pops out from the spatial location (see Figure 2(b)). It can be seen how a popped window can completely occlude another important feature such as the landmark in this example. Finally, Figure 2(c) shows that our proposed method can bring gradual changes to the heights of different portions of the spatial map to prevent the landmark from being occluded.

3.1 Overview

Figure 2 suggests that the method needs to accomplish the following tasks:

1. Identify the region of the spatial map which is potentially occluded by the standing window since this is precisely the region to be elevated.



(a) (b) (c)

Figure 2: **Illustration of the main benefit of our technique: (a)** The spatial map with a placemark placed by the user. **(b)** A window appearing right in front of the landmark has occluded the landmark. The window is displaying the average water elevation of the selected region(red rectangle) over time. (The temporal axis is horizontal from left to right.) **(c)** Elevating certain regions of the map using our method has made the landmark visible again. **(Please watch the supplementary video for a complete demonstration.)**

2. Determine the amount of elevation for each spatial point within the region that needs to be elevated.

A pre-processing step precedes the above mentioned steps where the spatial grid is divided in to 10×10 rectangular regions, known as *Blocks*. Any window that appears as a response to the user must stand along the boundary of a block. Although our technique does not enforce this condition on the window positions, Section 4 will reveal its advantage. The user can freely control the resolution of the grid of blocks which determines the size of the pop-up window.

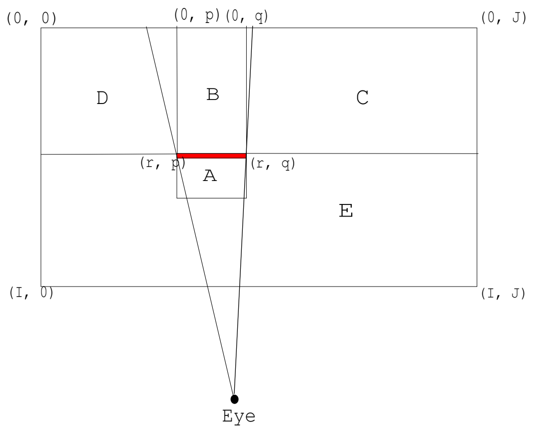


Figure 3: **Potentially occluded regions for 2D map:** When an infobox appears at the boundary of block A, regions B, C and D are potentially occluded and hence, needs to be elevated. Region E is unaffected.

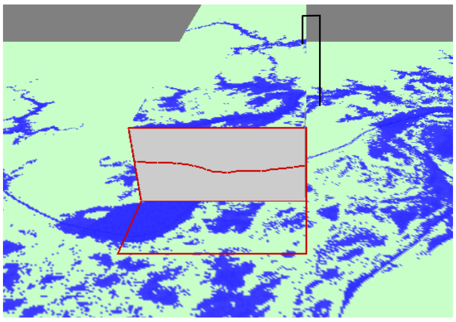
Our technique expects the following course of user interaction to happen when the default view is a 2D cartographic map. As the user clicks on one of the aforementioned blocks, a rectangular window, perpendicular to the 2D map, pops up in 3D. From now on, *Infobox*, a general name, will be used to refer to such windows. To enable viewing of the window, the 2D map automatically tilts backward so that its bottom end comes towards the user and the top end moves away (Figure 2(b)). An infobox always stands along the top boundary of the corresponding block. In Figure 2(b), the red rectangle on the map encloses the block corresponding to the infobox.

3.2 Identifying Occluded Regions

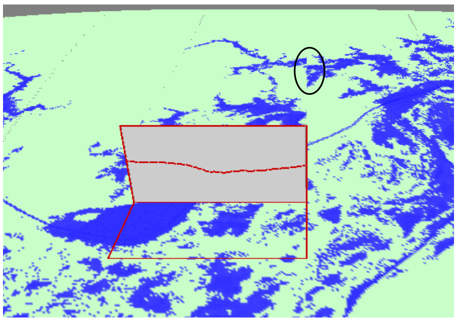
Figure 3 demonstrates how an infobox (the thin red rectangle) of adequate height standing along the boundary of block A can totally occlude region B and partially occlude regions C and D, when looked at from the bottom (or, from the front when the map tilts). Region E is unaffected. The exact amount of occlusion varies depending on the location of the infobox and the viewpoint. In general, the occlusion in regions C and D are expected to be much less than that in region B. Hence, imitating the approach described in [CS09], only region B could be elevated to the height of the infobox, to obtain a visualization like Figure 4(a). But unlike treemap which consists of discrete rectangles, geographic map changes continuously and an abrupt change of height may obscure geographic features, for example - river channels, lakes in this case.

3.3 Elevation of Identified Regions

The motivation of this step is to increase the heights of the potentially occluded regions smoothly possible when an infobox of height h appears. We have



(a)



(b)

Figure 4: **Distortion of the spatial map: (a)** Abrupt change of height of the spatial map can badly distort geographic features. The black line connects the two ends of a river channel which has broken due to abrupt lift of height. **(b)** Our technique, used with appropriate parameter values, causes less distortion of the same feature (circled in black).

employed different mathematical functions for different regions. We have made the elevation of region B (Figure 3) to vary between h and $h \times f$, where f is a fraction. The points closer to the infobox spanning from (r, p) to (r, q) are assigned higher elevation, which gradually falls down toward the other end (spanning from $(0, p)$ to $(0, q)$). Second, The maximum height of region D is made to be $h \times f$ at $(0, p)$ and the height gradually falls down to zero (in both horizontal and vertical directions in the Figure), as one moves away from that point. The height of region C follows a similar pattern of distortion, with the highest point at $(0, q)$. This formulation makes a smoothly lowering surface (region B) and two smoothly growing ones (C and D) meet at an intermediate height $h \times f$ at points $(0, p)$ and $(0, q)$. Figure 4(b) highlights the same feature to have distorted to a lesser extent using our technique. Latter sections will reveal how our technique enables the user to control the distortion.

Formulation of Elevation: Let us denote a spatial map by M. Any point on M can be denoted as $p = (m_i, m_j)$, where i and j respectively corresponds to the vertical and the horizontal direction in Figure 3. All co-ordinates are relative to the upper left corner of M. Given that an infobox of height h appears spanning from (r, p) to (r, q), the elevation of the surface adapts according to the following set of equations. h_{ij} denotes the increment in elevation required for (m_i, m_j). D_1 is a decreasing function of the vertical distance from the line from $(r, 0)$ to (r, J). D_2 is a decreasing function which takes as input distances in both directions with reference to a fixed point.

$$h_{ij} = h \times (f + (1 - f) \times D_1(|m_i - r|)), p \in B \quad (1)$$
$$h_{ij} = (h \times f) \times D_2(m_i, |m_j - p|)), p \in D \quad (2)$$
$$h_{ij} = (h \times f) \times D_2(m_i, |m_j - q|)), p \in C \quad (3)$$

3.4 Adaptive Nature of Infobox

Self-adaptive Height: In our technique, the basic 2D map has to tilt backward and distort itself to make the infoboxes visible. During exploration, if the user somehow loses spatial awareness, he may want to view the map at its original 2D orientation. In such a case, when the user rotates back, the raised infoboxes should not stand out as they would rapture the original 2D map due to perspective projection; they should not hide abruptly either. To avoid both, the height of each infobox maintains proportionality to the angle of rotation of the spatial map [CS09]. As the user rotates the map by angle θ, the eight should be $\theta \times k$, where k is some constant. As a result, the height of the infobox increases gradually with rotation and decreases smoothly as the user rotates in the reverse direction. This is how our technique achieves smooth navigation between overview and detail.

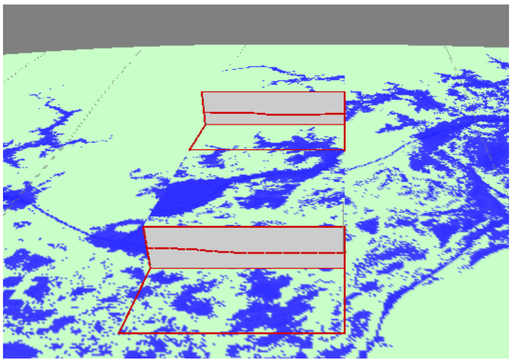
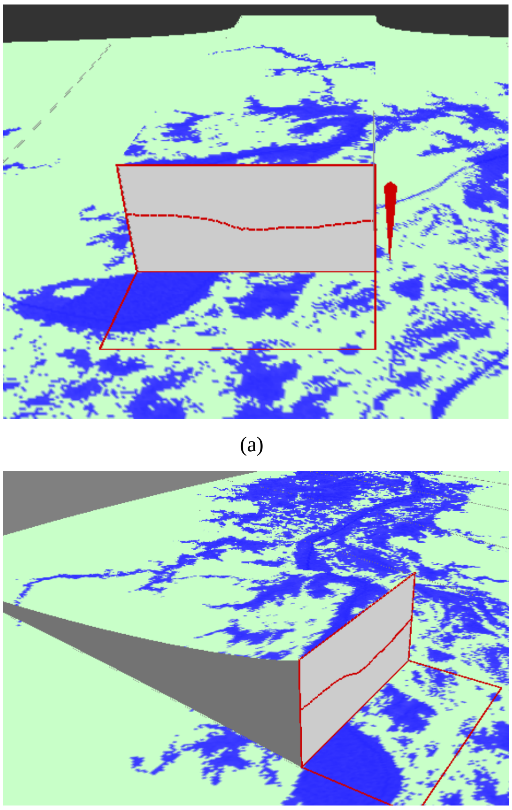


Figure 5: Spatial map with more than one infoboxes. The height of the infobox has been lowered to control the distortion of the spatial map.

Multiple Infoboxes: When more than one infobox appears, the same strategy can be applied to the subsequent ones and the final height of a spatial point results from addition of all the elevations due to the currently standing infoboxes. The computation of height of a spatial point is independent of the order in which the infoboxes appear or disappear (see Figure 5).

3.5 Extension for 3D Maps

The technique discussed so far works if the geospatial data is presented as a conventional cartographic map (Figure 1(a)), which would tilt backward to create usable space in 3D. But geospatial data are often represented on a virtual globe (like Figure 1(b)) and the user is provided with the freedom to move around in 3D space. In this case, when the user changes his viewpoint, the previously computed height map may not be effective anymore. To illustrate the problem, a landmark has been placed in a position (Figure 6(a)) which is not visible from a slightly sidewise rotated view (Figure 6(b)).



(a)

(b)

Figure 6: The self-adaptive elevation of Section 3.3 ceases to work when the user moves around in 3D space. A landmark which is visible from the viewpoint of **(a)** is occluded from the rotated viewpoint of **(b)**.

This problem necessitates the following enhancement of the proposed technique so that it can deal with spatial maps in virtual 3D environments.

3.5.1 Identifying Occluded Regions

In a way similar to what has been explained in Section 3.2, the spatial map is again divided into four regions (Figure 7(a)). Region B is the potentially occluded region. But region C (region D) also needs to adapt to reduce the spatial discontinuity with B which would otherwise be more prominent. Without loss of generality, we can assume the eye to be moving around the map along a hemisphere. Figure 7(b) shows the 2D projection of the hemisphere on the same plane as the spatial map. It can be noted that when the eye enters a different quadrant of the hemisphere (the circle in the diagram), the regions B, C, D and E must change position, even though their relative position with respect to one another remains the same.

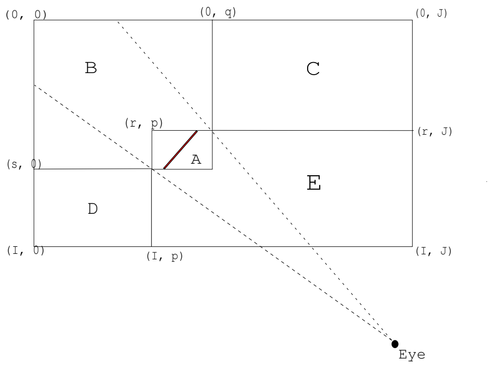
3.5.2 Elevation of Occluded Regions

When an infobox appears at the block denoted by A (Figure 7(a)), region B always includes the entirely occluded points and hence must be elevated. As before, the height of this region varies between h and $h \times f$. The corner of B which is adjacent to block A ((r, p) in the diagram) is elevated to h and the other three corners attain height $h \times f$. Elevation of regions C and D varies between ground level and $h \times f$. Even though these two regions are not occluded, they are elevated to gradually merge the discontinuity along the meeting plane between B and C and that between B and D. Figure 7(b) suggests that other than the few cases when the eye enters a new quadrant, no updating of the elevation is needed due to eye movement.

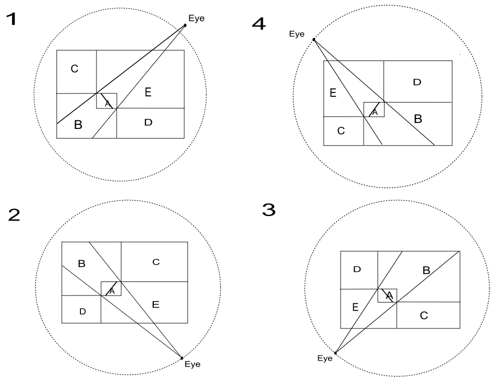
3.5.3 Orientation of the Infobox

The orientation of the infobox needs to be updated with every angular change of the eye position though. As can be observed in Figure 7, the infobox no longer stands along the top end of the corresponding block, it rather orients in such a way that each it always faces the viewer. The update required for this is confined to the block which contains the infobox and hence, not computationally expensive.

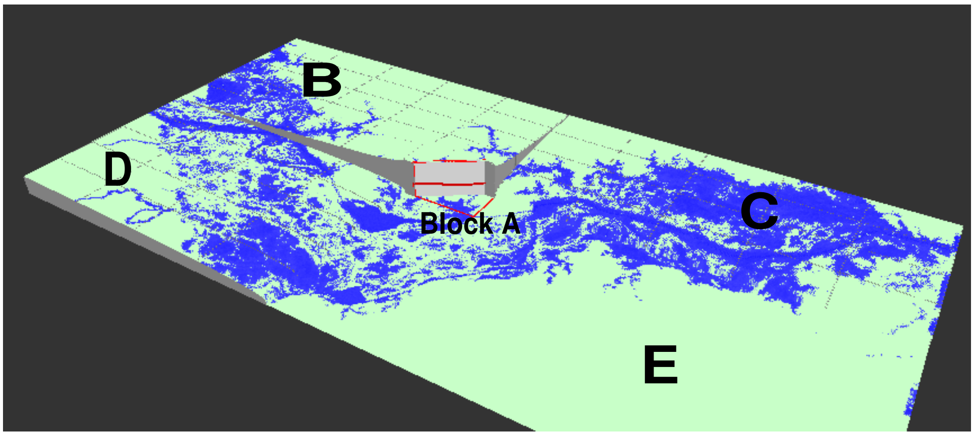
The view direction for an infobox is computed as the vector $\vec{v} = \vec{c} - \vec{eye}$, where $\vec{c}$ is the position vector of the center of a block A. 2D projections of all the vectors on the plane of the spatial map suffice for the computation. The infobox is oriented along



(a) Re-defined subdivision of the spatial map based on the occluded regions in case of 3D representation.



(b) The location of the regions must change when the eye enters a different quadrant. The elevation of each spatial point is re-computed accordingly.



(c) The spatial map with an infobox from bird's eye view. The conceptual divisions have been annotated. The infobox can be seen to face the observer.

Figure 7: Figures related to the extension of the idea in case of spatial maps represented as virtual globe

the direction orthogonal to the view direction (Figure 8). Now, for any point e contained in A, the angle between $\vec{e} - \vec{c}$ and $\vec{v}$ determines if e is occluded by the infobox. If occludes, the point is elevated to the height of the infobox. Point f in Figure 8 is an example of a point which is not occludes and so, does not require elevation.

4 Design Choices

Besides the central idea already explained, the implementation of the technique needs to deal with a few more important issues.

4.1 Functions and Parameters

In our technique, functions have been employed to smoothly change the height of the occluded portions. Figure 9(a) illustrates the gradual rise and decay of two surfaces, which finally merge at an intermediate height between 0 and h. The choice of the function and its parameters is important, since it determines the extent of the occlusion and the distortion of the spatial map. For example, in Figure 9(a), the three curves in the upper half represent $D_1 = (1-x^p) \times (1-x^p)$ with three different values of p. This is the curve that we have used to control the gradual decrease of height in region B (Section 3.3). Similarly, the gradual rise of height in regions C and E has been controlled by $D_2 = x^q$, represented by the three curves in the lower half. Any one from the upper half can be used in pair up with one from the lower half.

It can be noted in Figure 9(b) that the curve A1 can remove less occlusion than A3, whereas A3 reduces more occlusion at the cost of creating a wider discontinuity between A3 and its counterpart from the lower half. Similarly, B3 causes much less distortion of the spatial map than B1, but leaves a wide gap between

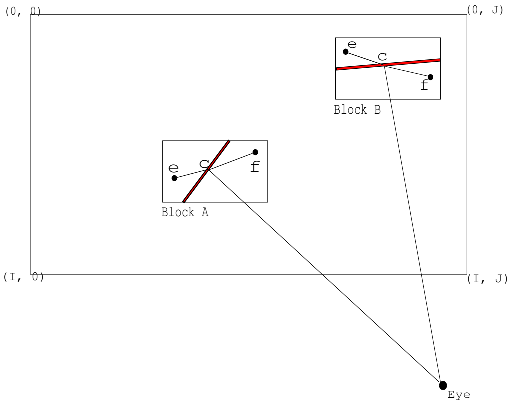
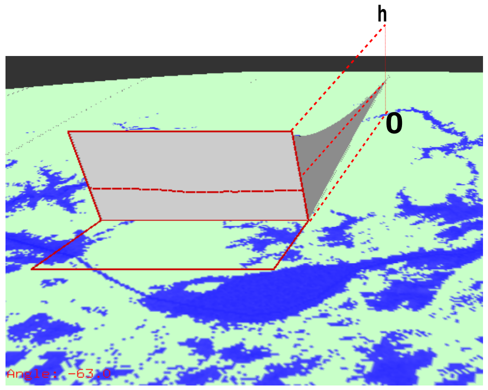
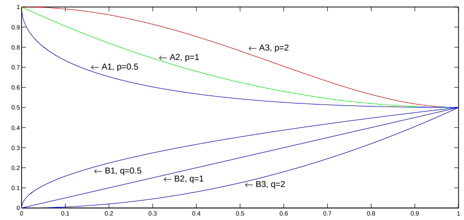


Figure 8: The figure shows two infoboxes oriented differently. Both are looking at the eye position. The orientation needs to be re-computed with every angular change of eye location. For both the infoboxes, e is an occluded point and f is a visible point. c is the center of the infobox.



(a)



(b)

Figure 9: The decay function and its parameters control the amount of occlusion and spatial distortion. **(a)** Illustration of the decay function. The gradually lowering surface is controlled by D_1 and the other one by D_2. **(b)** 1D decay function along with different parameters.

the two regions. Figure 10 shows how the function parameters controls the extent of occlusion and the appearance of a feature. The parameters exert similar influence on occlusion and distortion for the 3D maps as well.

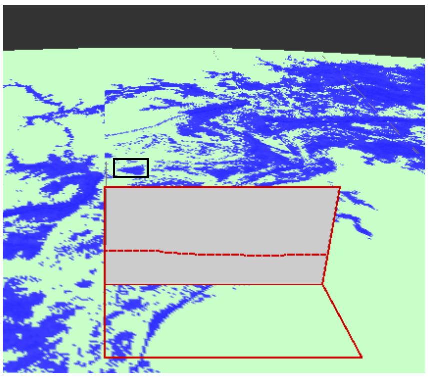
4.2 Division into Blocks

The final height of each spatial point on the map depends on its distance from the infoboxes, but not on the block (defined in Section 3.1) it belongs to. Hence, our method is independent of the existence of blocks and as shown in the Figure 11(left), an infobox can be raised from any arbitrary point p. Suppose the infobox spans up to q, the affected regions are elevated accordingly. Now, if the user wants to pop up another infobox at point r, it can be seen that the new infobox has to cut through line wq, which divides two surfaces at different heights. In other words, the two bases, r and s of the new infobox, already lie at two different elevations. One solution can be to restrict the length of the new infobox to the point (in this case t) where it spans over a different elevation zone. We have adopted another solution, which is to attach the location of an infobox to its adjacent block so that an infobox never stretches beyond its own block (Figure 11(right)).

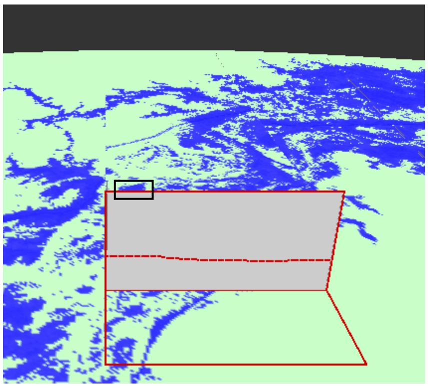
4.3 Dimension of the Infobox

Good aspect ratio of the infobox is required for clarity of the information displayed on it. First, the user can adjust the proportionality constant that determines its height for a given position of the spatial map. Second, for 2D maps, the width of the infobox is always equal to the width of a block (Figure 3). Hence, the resolution of the blocks can be adjusted by the user to obtain the intended width. The height of the infobox can be changed freely as long as its aspect ratio and the map's distortion are within acceptable limits.

For the 3D virtual globe scenario, the orientation and hence, the aspect ratio of the infobox changes with every bit of rotation, because it rotates always keeping the two ends of its base on the perimeter of the block. Hence, the aspect ratio of an infobox, contained in a block of width w and length l depends on its base length, which can vary between w and $\sqrt{w^2+l^2}$. It can be noted that for a square block, the base length always remains between w and $\sqrt{2}\times w$. Hence, to avoid major change in aspect ratio due to eye movement, the division of the spatial map should be done in such a way that each block is as close as possible to a square.



(a)



(b)

Figure 10: **(a)** When D_1 with a higher p is used as function, the resulting image clearly shows the circled blue area, possibly a water body. **(b)** The feature can get partially or totally occluded when the parameter value is lowered.

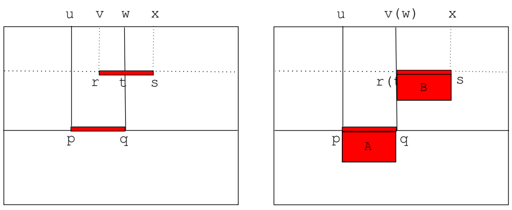


Figure 11: **Subdivision into Blocks:** The reason behind subdividing the spatial map into rectangular blocks is illustrated.

5 Application

Our main focus so far has been to create the infoboxes. The images presented up to this point have utilized the infobox for conventional representation of time series data related to the spatial location. This section demonstrates two more ways of utilizing the infobox to emphasize that its use is not limited to visualizing any particular form of data.

5.1 Spatio-temporal Visualization

From the perception point of view, human eyes can compare two things most effectively when one is placed next to the other as a mirror image. The symmetric shape formed by the two objects together makes the task easy. In the context of spatio-temporal data, an analyst may be interested in comparing a region to its temporal snapshot from a different time step to know how certain properties of the region will change over time. However, two difficulties can arise here:

1. The data fetched from the other time step cannot be overlaid on the current map since that will make comparison impossible.
2. If the fetched data is displayed in a traditional window, the window is located at a distance from the region of interest which is not desirable for comparison purpose.

This is where our technique, which always keeps the infobox coupled to its spatial root, can be useful. We employ infobox as a mirror-like display which contains the spatial data from a different timestep. Figure 12 is an example where the infobox is the snapshot of the selected block from 50 timesteps ahead. Furthermore, different infoboxes can display different timesteps (Figure 12) and can be controlled or animated independently. (The accompanying video provides an application in more detail.)

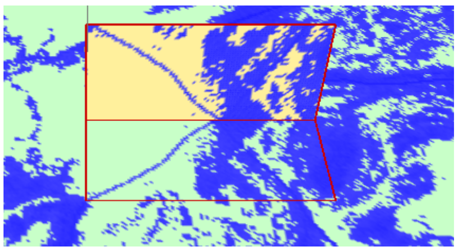


Figure 12: **Spatio-temporal visualization:** The infoboxes can display temporal snapshots from other timesteps right beside the current map of the same region. The map from the other time step has been flipped as if it is the mirror image of the original. This helps comparison from cognitive point of view.

Since the actual map and the corresponding one in the infobox are orthogonal to each other, certain comparisons such as matching the shape of a feature, may be difficult because of perspective viewing. Many other tasks, such as checking the presence of a current feature in a distant time step and appearance of new features can be easily performed with the presented display.

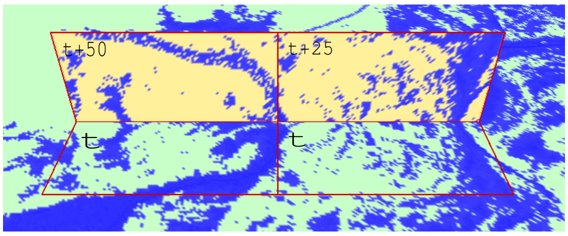
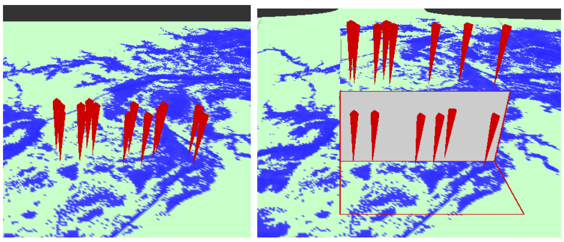


Figure 13: Different infoboxes can display different time steps to allow analysis on the local scale. The red texts denote the time step of the surrounding block.

5.2 Visual Clutter Reduction

Figure 1(b) stands witness to the fact that visual annotations like placemarks themselves can clutter a map. Google Earth scatters the cluttered placemarks and places a representative single dot at a location close them. Multiple arrows diverge from the dot to connect to the shifted placemarks. In our technique, for a region populated by placemarks (Figure 14(a), the block resolution can be adjusted to let a block boundary pass through the cluttered region. Then, an infobox with a blank face can be raised only to deliberately create a height difference. Figure 14(b) displays the dispersed placemarks, even though each one is still connected to its spatial root.



(a) (b)

Figure 14: **Infobox employed to clutter reduction: (a)** Visual clutter of icons. **(b)** The deliberately introduced height difference reduces clutter, keeping the spatial connection of each placemark intact.

6 Conclusion and Future Work

We have proposed a self-adaptive technique for reducing occlusion which is a serious concern in geovisualization. Our method has been tested with a moderate size spatio-temporal data set of Amazon River Basin. We have demonstrated a couple of examples of how our technique can be applied to clutter reduction and comparative analysis of spatio-temporal data.

Our method has a few significant differences with traditional cartographic map with windows to display details:

1. Unlike windows that can be moved and resized, our infoboxes are always rooted at the spatial map to provide better visual correspondence. The width and the height of the infoboxes can be changed by the user, but the change takes effect uniformly on all of them. A smaller infobox at one location cannot co-exist with a larger one at another location in the current framework. Apparently, adaptive sub-division of the spatial map into blocks of different sizes can enable infoboxes of variable sizes.

2. Because of the small dimension of infoboxes, a user may need to zoom in to inspect the details presented in the infobox. However, zoom-in and zoom-out being the most frequently used interaction for geospatial data, which often comes in multiple resolutions, this is not a burden to the user after all.

These differences do not limit the utility of our method since the infoboxes can co-exist with traditional dispplays such as landmarks, push-pins and windows. The technique can be employed when needed, preserving the other placemarks and visualizations. We intend to explore the potential of using infoboxes as mirror-like displays for visual analysis of geospatial as well as scientific data.

References

[AA05] Gennady Andrienko and Natalia Andrienko. Visual exploration of the spatial distribution of temporal behaviors. *Information Visualisation, International Conference on*, 0:799–806, 2005.

[BDW+08] Thomas Butkiewicz, Wenwen Dou, Zachary Wartell, William Ribarsky, and Remco Chang. Multi-focused geospatial analysis using probes. *IEEE Transactions on Visualization and Computer Graphics*, 14(6):1165–1172, 2008.

[Bra97] R. Brath. Metrics for effective information visualization. In *INFOVIS '97: Proceedings of the 1997 IEEE Symposium on Information Visualization (InfoVis '97)*, page 108, Washington, DC, USA, 1997. IEEE Computer Society.

[Bre94] Cynthia A. Brewer. Color use guidelines for mapping and visualization. *Visualization in Modern Cartography*, pages 123–147, 1994.

[Cor09] Microsoft Corporation. Microsoft bing website http://www.bing.com/maps, 2009.

[CS09] A. Chaudhuri and Han-Wei Shen. A self-adaptive treemap-based technique for visualizing hierarchical data in 3d. In *Visualization Symposium, 2009. PacificVis '09. IEEE Pacific*, pages 105–112, April 2009.

[CWK+07] Remco Chang, Ginette Wessel, Robert Kosara, Eric Sauda, and William Ribarsky. Legible cities: Focus-dependent multi-resolution visualization of urban relationships. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1169–1175, 2007.

[DB07] Jason Dykes and Chris Brunsdon. Geographically weighted visualization: Interactive graphics for scale-varying exploratory analysis. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1161–1168, 2007.

[DBN06] Danny Dorling, Anna Barford, and Mark Newman. Worldmapper: The world as you've never seen it before. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):757–764, 2006.

[ED06] Geoffrey Ellis and Alan Dix. The plot, the clutter, the sampling and its lens: occlusion measures for automatic clutter reduction. In *AVI '06: Proceedings of the working conference on Advanced visual interfaces*, pages 266–269, New York, NY, USA, 2006. ACM.

[ESR09] ESRI. Arcgis website, http://www.esri.com/software/arcgis, 2009.

[ET08] N. Elmqvist and P. Tsigas. A taxonomy of 3d occlusion management for visualization. *Visualization and Computer Graphics, IEEE Transactions on*, 14(5):1095–1109, Sept.-Oct. 2008.

[GTWH02] Mark Gahegan, Masahiro Takatsuka, Mike Wheeler, and Frank Hardisty. Introducing geovista studio: an integrated suite of visualization and computational methods for exploration and knowledge construction in geography. *Computers, Environment and Urban Systems*, 26(4):267–292, 2002.

[Inc09] Google Inc. Google eath website, http: www.earth.google.com, 2009.

[KW04] T. Kapler and W. Wright. Geotime information visualization. In *Information Visualization, 2004. INFOVIS 2004. IEEE Symposium on*, pages 25–32, 0-0 2004.

[Mac01] MacEachren. Research challenges in geovisualization. *Cartography and Geographic Information Science*, 28:3, 2001.

[NAS09] NASA. Worldwind website, http://www.worldwind.arc.nasa.gov, 2009.

[PGSF04] John Plate, Anselm Grundhöfer, Benjamin Schmidt, and Bernd Fröhlich. Occlusion culling for sub-surface models in geo-scientific applications. In *VisSym*, pages 267–272, 351, 2004.

[PSKN06] Christian Panse, Mike Sips, Daniel Keim, and Stephen North. Visualization of geo-spatial point sets via global shape transformation and local pixel placement. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):749–756, 2006.

[QWC+09] Huamin Qu, Haomian Wang, Weiwei Cui, Yingcai Wu, and Ming-Yuen Chan. Focus+context route zooming and information overlay in 3d urban environments. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):1547–1554, 2009.

[SFdOL04] Milton H. Shimabukuro, Edilson F. Flores, Maria Cristina F. de Oliveira, and Haim Levkowitz. Coordinated views to assist exploration of spatio-temporal data: A case study. *Coordinated and Multiple Views in Exploratory Visualization, International Conference on*, 0:107–117, 2004.

[SRd05] Poonam Shanbhag, Penny Rheingans, and Marie desJardins. Temporal visualization of planning polygons for efficient partitioning of geo-spatial data. In *INFOVIS '05: Proceedings of the Proceedings of the 2005 IEEE Symposium on Information Visualization*, page 28, Washington, DC, USA, 2005. IEEE Computer Society.

[TGC03] M. Trutschl, G. Grinstein, and U. Cvek. Intelligently resolving point occlusion. In *Information Visualization, 2003. INFOVIS 2003. IEEE Symposium on*, pages 131–136, Oct. 2003.

[TSWS05] C. Tominski, P. Schulze-Wollgast, and H. Schumann. 3d information visualization for time dependent data on maps. In *Information Visualisation, 2005. Proceedings. Ninth International Conference on*, pages 175–181, July 2005.