Multilayer Hybrid Visualizations to Support 3D GIS

Stephen Brooks [a,*], Jacqueline L. Whalley [b]

[a] *Faculty of Computer Science, Dalhousie University, Halifax, Nova Scotia, Canada*
[b] *Department of Computing and Mathematical Sciences, Auckland University of Technology, Auckland, New Zealand*

Abstract

In this paper we present a unique hybrid visualization system for spatial data. Although some existing 3D GIS systems offer 2D views they are typically isolated from the 3D view in that they are presented in a separate window. Our system is a novel hybrid 2D/3D approach that seamlessly integrates 2D and 3D views of the same data. In our interface, multiple layers of information are continuously transformed between the 2D and 3D modes under the control of the user, directly over a base-terrain. In this way, our prototype system is able to depict 2D and 3D views within the same window. This has advantages, since 2D and 3D visualizations can each be easier to interpret in different contexts.

In this work we develop this concept of a hybrid visualization by presenting a comprehensive set of capabilities within our distinctive system. These include new facilities such as: hybrid landmark, 3D point, and chart layers, the grouping of multiple hybrid layers, layer painting, the merging of layer controls and consistent zooming functionality.

Keywords: 2D and 3D visualization; geographical information systems; hybrid display; interactive.

* Corresponding author. Tel.: + (1) 902 4942512; fax: + (1) 902 492-1517 5.
E-mail addresses: sbrooks@cs.dal.ca, jwhalley@aut.ac.nz.

1. Introduction

In the past, most geographical information systems (GIS) were limited to providing visualizations of data in two dimensions (2D GIS). Currently, GIS research and development still lies largely in this traditional map-based approach. While 2D GIS can be used to perform numerous spatial analyses and applications, visualizations are generally limited to viewing either individual GIS layers or the results of data queries (Zlatanova et al., 2004).

We relate to our world in three or more dimensions, which suggests that some types of data may be more readily visualized and analyzed in 3D. Until recently 3D GIS were not practical due to delayed responses to user interaction (Faust, 1995). But 3D GIS have now become a reality due to advances in computer graphics such as fast graphics processing units (ATI™, 2007; NVIDIA®, 2007) and efficient terrain visualization algorithms (Hoppe; 1996; Lindstrom et al., 1996).

However, direct 3D analogues to 2D GIS are not ideal solutions because they suffer from several shortcomings. One shortcoming is that while interactive computer graphics applications, such as 3D maps, have been used for the exploration, presentation and manipulation of geographical data, gaining insight from 3D spatial datasets can be particularly challenging. Moreover, a high data density can make it difficult to view all the data at once due to the self-occlusion of the data. Self-occlusion can be particularly acute when attempting to present and interpret multivariate data in a meaningful way. Additional difficulties can arise in viewing information in 3D GIS when a terrain is hilly because the elevated regions in the terrain can occlude more distant data. Finally, there is the challenge of establishing how best to simultaneously visualize different types of data in 3D.

Studies have shown that 2D views are often used to establish precise relationships, while 3D views help in the acquisition of qualitative understanding (Springmeyer et al., 1992). Both dimensionalities of view therefore have distinct advantages and it would be ideal if the benefits of both 2D and 3D could be incorporated into the same system. Although many existing commercial 3D GIS systems have 2D views they are typically isolated from the 3D view in that they are presented in a separate window.
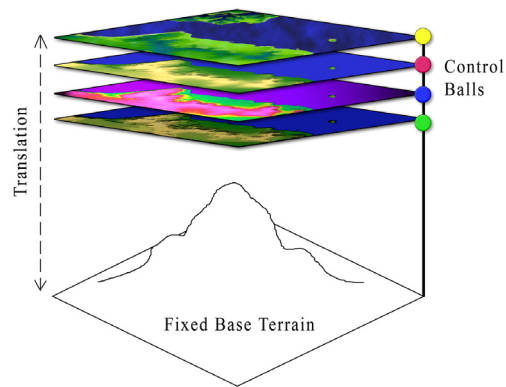

Fig. 1. Multiple 2D/3D layers in our hybrid display over a 3D base terrain. The vertical translation of each layer is set with its associated control ball.

Our hybrid system seamlessly integrates 2D and 3D views of the same data and allows the user to view 2D data in direct relation to the 3D view within the same window. Layers are visualised in a combined overlay representation where multiple heterogeneous layers of information are continuously transformed between the 2D and 3D over the base-terrain (see Fig. 1). The combination of geo-referenced data and aspatial attributes, with integrated 2D and 3D modes of operation, is designed to enable the extraction and visualization of complex spatial relationships. It is intended that this system will provide an interface that allows the exploration and understanding of structures, patterns and processes reflected in both 2D and 3D data.

2. Related Work

Recently 3D GIS have received considerable attention and the literature surrounding the area is growing. In 2002, Zlatanova et al. produced a survey of mainstream GIS software. They reviewed a number of systems including: ArcGIS (ESRI, 2007), Imagine Virtual GIS (ERDAS, 2007), PAMAP GIS Topographer (PCIGEOMATICS, 2007) and Geomedia Terrain (Integraph, 2007). They found that some initial steps forward have been made in terms of the visualization and animation of 3D spatial data and spatial objects, but that 3D GIS still often lack the basic GIS functions.

Further examples of existing 3D research prototypes include Terrafly (Rishe et al., 2004), GeoVR (Huang and Lin, 1999) and TerraVisionII (Reddy et al., 1999). One noteworthy system, called GeoTime (Kapler and Wright, 2005) proposes an interesting solution to the problem of integrating timeline events into interactive GIS. The Adelaide 3D GIS planning model (2007) is an example of a 3D GIS which provides 3D visualisations of social and environmental data within a 3D cityscape. Two other interesting applications are called GeoZui3D (Ware et al. 2001) and VGIS (Köller, 1995). GeoZui3D is a 3D marine GIS that supports real-time input and draped imagery; while VGIS is an integrated global GIS and visual simulation system.

Recently, Stoter and Zlatnaova (2003) revisited the current status of 3D GIS and postulated that 3D GIS is merely at a point where 2D GIS was several years ago. While these perspective renderings are generally accepted as showing the relationships of the GIS data to the natural terrain, in reality they have limited the efficacy of the perspective images to 'show and tell' type applications (Stoter and Zlatanova, 2003). In order to utilize the relationship of data to its natural terrain additional information must

be provided along with the perspective image in order to make measurements of size and shape. Furthermore, this type of representation only permits the user to view map layers as a single entity rather than being able to visualize the layers in a combined overlay representation.

The applications for effective GIS visualisations are vast and include environmental analysis and modelling, flood models, geological models and urban planning. As early as the late 1980's the development and the role of 3D-GIS for urban planning was being investigated (Köeninger and Bartel, 1998). More recently, GIS has come into focus in the field of sustainable urban design. Sustainable urban design has become a major challenge for urban planners (Yao et al., 2006) and there is a requirement for computer-based tools which allow planners and designers to evaluate and analyze the environmental impacts of their proposed development. The benefits and limitations of computer-based visualisation of a 3D cityscape for urban planning in the context of the ADEQUA project in Portugal have been reported (Molines et al.,2006).

For some urban planning tasks such as noise, flood and pollution modelling 3D GIS is believed to be beneficial (Stoter and Zlatanova, 2003). However 3D visualisations in urban planning and environmental impact modelling still face the same limitations as 3D GIS developed for other applications.

Further research is required to explore the possibilities and constraints of 3D GIS in order to move beyond the somewhat limited scope of providing a tool for terrain flybys and map-making. Indeed this is the definitive aim of the research presented in this paper. But gaining insight in 3D is complicated by the fact that a high data density makes it difficult to view all data at once since the data can self-occlude and elevated regions in the terrain occludes data further back. Attempts to overcome these problems,

in 3D GIS, usually involve displaying the terrain from several different viewpoints. Verbee et al. (1999) proposed a multi-view approach based on different types of visualization to support interaction with a 3D GIS. Their multi-view approach allowed interaction of GIS data within 3 different view types, namely: plan view, model view and world view. However, separate views introduce new problems since the integration and interpretation of the multiple views must occur in the mind of the user, thereby placing more demands on the user. Another issue is that the more views are displayed, the smaller each view must become.

We propose that by providing 2D-to-3D transitional layers we can overcome both the self-occlusion and terrain-occlusion issues. Our layering system also offers a convenient means of handling multiple heterogeneous sets of aspatial data under user control. Additionally, the system allows the user to temporarily set aside data that is not currently relevant. Our work builds upon 3D GIS visualizations and is somewhat related to a small number of prototype medical visualization systems.

Medical visualizations sometimes incorporate aspects of 2D and 3D views in some fashion. For example, with regard to MRI data, 2D slices can be combined with a 3D overview using one of two main approaches: clip planes and orientation icons (Tory & Swindells, 2003). In orientation icon systems (Fig. 2, right), the 3D overview and 2D details are in separate windows. Whereas, clip planes show slice details in their exact relative position to the 3D context (Fig. 2, left). This clip plane visualisation technique is the approach that is most similar to our hybrid display. But as will be seen, these systems that provided the initial inspiration for our combination display are in the end quite different from our hybrid GIS.
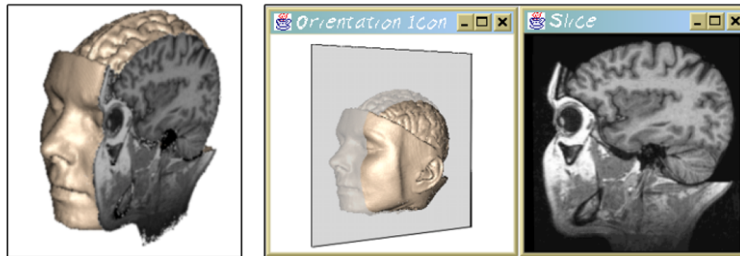
Fig. 2. Medical clip-planes (left) and orientation icons (right).

In GIS, little research and development has been undertaken with respect to hybrid displays. This is despite the fact that this type of display allows the dual strengths of 2D and 3D visualization to be exploited (Troy et al., 2004). We achieve a combination display with a unique system of in-place transitional layers. The 3D view and the hybrid 2D/3D layer system are rotated and translated synchronously so that they maintain an identical orientation and position.

What follows is an in-depth discussion of our hybrid display and its implementation. In section 3 we discuss the essential features of our system. We follow this in section 4 with a discussion of the aggregate grouping of layers and offer unified control mechanisms for the grouped constituents. In section 5 we then introduce several new types of 3D layer content including: Landmark Layers, Chart Layers and 3D Point Layers. Section 6 discusses the addition of a zooming facility, both in terms of usage and implementation. In the closing sections we propose a set of future directions and conclude with a summary of contributions.

3. Fundamental Features of the Hybrid 2D/3D GIS

Our initial prototype system offered a rudimentary a set of features that provided the core hybrid functionality for the manipulation of layers as reported in a preliminary conference paper (Brooks & Whalley, 2005). Therein we described the key concepts

upon which our system is built including our unique layer system that allows layers to be continuously transitioned between 2D and 3D representations, and the base terrain upon which it rests.

3.1   Rendering of the Base Terrain

The base terrain consists of a regular grid of Digital Elevation Model (DEM) data. Each elevation point in the DEM is interpolated from a spot height measurement or contour data (Yu, 2004) as shown in Fig. 3. There are two general approaches available for constructing a 3D model from DEM data. The first approach is to use TIN (Triangulated Irregular Network) as a global surface model (Hoppe 1996) and the second is to employ direct processing of all points as a regular grid (Lindstrom et. al. 1996). At the present phase of development we have chosen to use the direct processing approach.
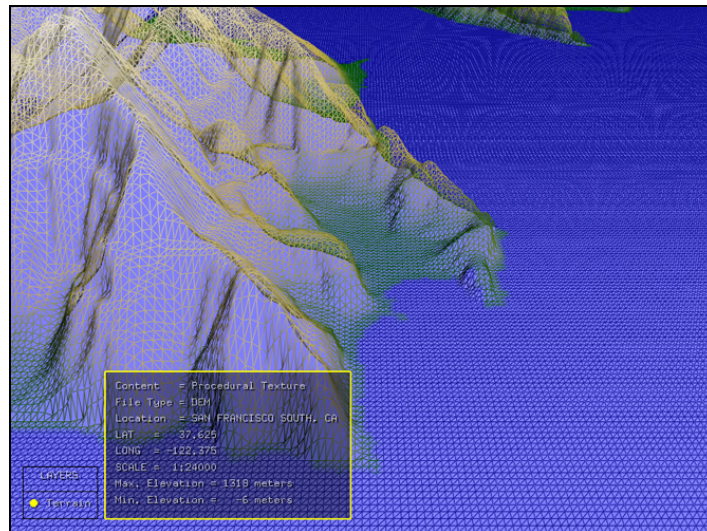

Fig. 3: A regular grid DEM shown in wire-frame mode.

In order to produce an accurate final appearance, we have the option of integrating high resolution satellite images as textures on our wire-frame mesh. However, care must be taken when registering satellite imagery onto DEM data. In

addition, it may be necessary to mosaic multiple satellite images due to imperfections on individual images, particularly for cloudy regions. In the absence of (or in addition to) good satellite images, a procedural texture can be generated based on elevation values. Our system generates a standard texture with colours ranging from blue, below zero elevation, up through brown, green and white. In addition, there is a modest degree of fractal noise that is added to the texture in order to avoid an overly sterile appearance.

Much of the data we used to develop our system was acquired from the San Francisco Bay Area Regional Online Database and the U.S. Geological Survey data repository. The terrain data is extracted from the DEM format and vector data from the DLG format.

Furthermore, Troy (2004) found that shadows play an important role in conveying information to the user of 3D information systems. They reported that displays with shadows had lower difficulty ratings than 3D rotated or 2D displays largely due to that fact that 3D rotated displays made it difficult to estimate height. They concluded that displays with shadows can result in effective 3D displays but only if the light is placed in a very specific position in relation to the object of interest. In our GIS, the user has a choice with regard to light positioning and orientation. The light position can either be fixed relative to the camera position or fixed relative to the base-terrain. Also, the light can either be a spotlight or directional.

3.2 The Hybrid 2D/3D Layer System

What makes our visualization approach unique is the multiple layers of information that can be continuously raised or lowered by the user directly over the base-terrain. During

elevation, the transitional layers maintain their original geographical (latitude/longitude)

position relative to the base terrain. When translated skywards, the height values are

gradually flattened into a completely flat 2D map layer (Fig. 4a-c).
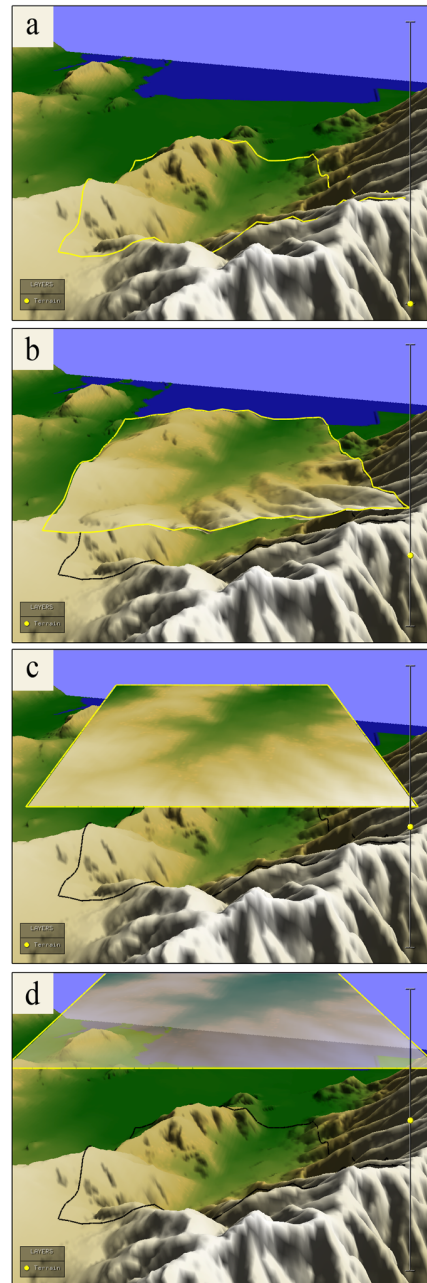

Fig. 4. The layer is shown gradually rising from image *a* to image *d*. As the layer rises it
morphs into a flat plane. Image *c* shows the layer completely flat. Above flat level, the
layer becomes increasingly transparent as shown in *d*.

Each layer's translation is set with an associated user-positioned control ball. These control balls are shown to the right in Fig. 4 and diagrammatically in Fig. 1. There is a linear mapping between the movement of the control ball and the vertical translation of the associated layer, with the maximum and minimum values of the control ball slider mapped directly to the maximum and minimum height position of the layer. The color of the control ball is unique for each layer. In addition the layer itself is trimmed along its edge with the same color as its associated control ball. In Fig. 4 the layer and control ball are color coded in the same color. This application of consistent color coding finds support in user interface design principles (Hix & Hartson, 1993).

When a layer is high above the terrain, it flattens out into a conventional 2D map. This allows the user to analyze information in 2D and also provides a way of viewing information that may have been hidden by near elevations in the 3D view. It also allows to user to move information out of view when he or she is not interested in the content of a particular layer. This is achieved by simply raising the layer to the top of the screen (Fig. 4d). At a certain height the layer becomes flat, and above this height the layer becomes increasingly transparent, until all that is visible is the colored outline that trims the layer's edge. This layer trim is not affected by the changes in transparency of the layer content.

In order for a transitional hybrid layer system to be useful the user should be able to mentally map a flattened 2D layer onto the 3D terrain that it is residing over. To aid this, a ground level shadow of the elevated layer system is provided which indicates the correlation between the raised area of data in the 2D/3D layers and the 3D base terrain.

Our original system offered a number of thematic layers in addition to the basic DEM terrain layer. These included a number of vector line data layers (such as road and hypsography layers) and color coded texture layers (such as an atlas layer). The layer's texture can either be a satellite image, a procedurally generated texture based on elevation data, or a DLG (ArcView™ Digital Line Graph) vector data layer pre-computed onto the texture mapped geometry. Fig. 5 shows five layers shown over a San Francisco DEM terrain. Two of the layers have been raised to the flattening level. A transparent black-line hypsography layer is placed slightly above an atlas-coloured texture layer. The remaining 3 levels (terrain, railroads, and roads) have been raised out of the way to the top of the screen. These layers have become completely transparent at this level except for their color coded edge trims. Each of these layers requires its own meta-data be available in an auxiliary pop-up legend (Fig. 6). In section 4 we extensively expand the range of possible hybrid layers; in particular we will allow a variety of 3D objects to be embedded in layers.
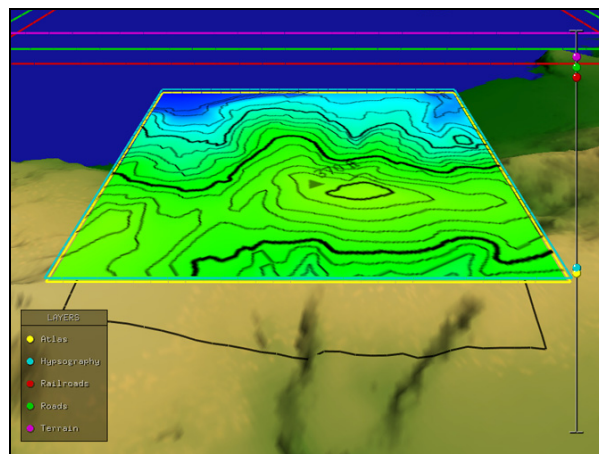

Fig. 5. Five layers over a DEM terrain. Two layers have been raised to the flattening level with the black-line hypsography layer slightly above the atlas-coloured texture layer. The remaining three levels have been raise to the top of the screen.
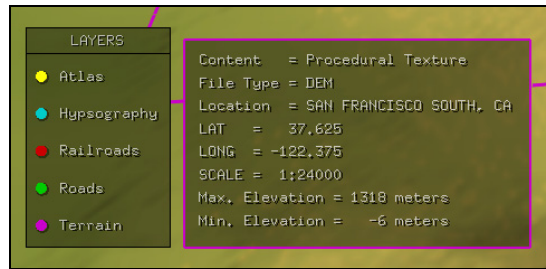
Fig. 6. Text based meta-data pop-up legend for a layer.

3.3   Navigation Tools

Gaining insight in 3D GIS can be complicated by the fact that it is easy to get "lost" or

disorientated. A number of navigational methods have been proposed to help orientation

and comprehension in a virtual environment (Darken & Silbert, 1993). In addition,

navigational metaphors have been proposed to aide a user's direction-finding in a 3D

terrain. One such metaphor is the human walker (van der Linden et al., 2004).

In our system we integrate several standard navigation capabilities expected of

the traditional 2D map-based approach. The set of navigational aids integrated into our

GIS system include: a directional compass (Fig. 7), a mini-locator overview map (Fig.

8), and a dialog box that allows the user to enter a latitude and longitude. The mini-map

shows the current location of the layering system as an orientated red square and also

allows the user to "jump" between locations on the terrain by directly selecting a

location on the mini-map. This positions the camera at that location and also stores that

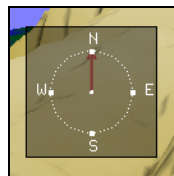location in a list for later use.

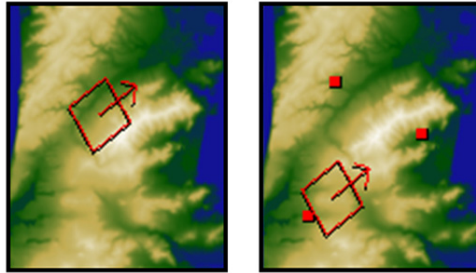Fig. 7. Semi-transparent directional compass.

Fig. 8. Mini-map with indicator box and arrow (left). A series of jump sites (right).
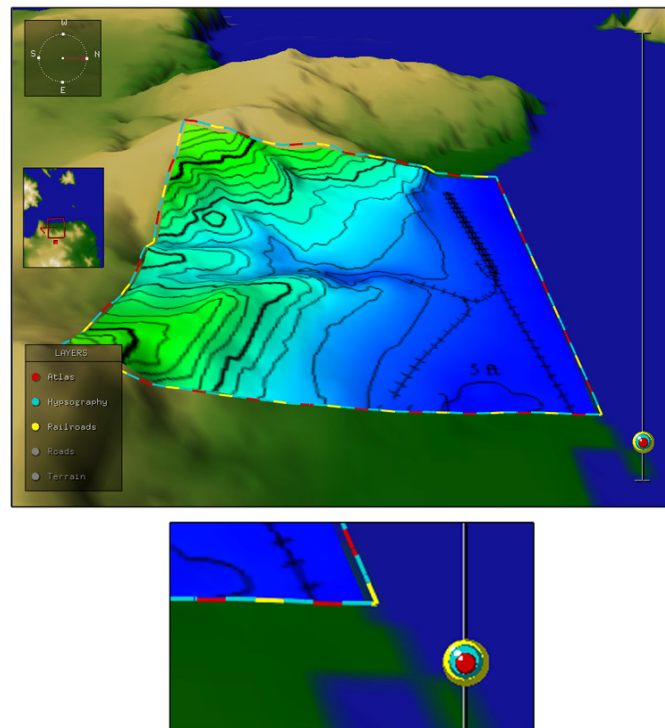

Fig.9. Top: Multiple thematic layers grouped into a single unit. Bottom: close-up of multiple control balls in an onioned state. Also note the layer edge trim pattern which has a matching color set.

4. Layer Grouping

Our layering system offers a convenient means of handling multiple heterogeneous sets of aspatial data by separating the data content into hybrid layers. Each layer height can be controlled via its associated control ball. A further extension, which we have

implemented, to the layer system is the ability to group two or more layers into a single entity. The user can form a new layer group simply by raising (or lowering) layer *A*'s control ball onto the control ball of layer *B*. Layers can also be added to an existing group by dragging additional control balls to the same height. An example of three grouped layers is shown in Fig. 9, Top. The layers are: atlas, railroads and hypsography.

When multiple layers are grouped it is important to provide a visual indication of the grouping and of the individual member layers in the group. We achieve this in two complementary ways. Firstly, the edge trim that surrounds each layer is adapted to indicate the grouped state of the layers. The new grouped state edge trim is formed by taking each of the representative colors for each of the layers and incorporating these colors into an alternating dash pattern. Secondly, we alter the control balls to form an onion-like configuration. We display the combined control balls as if they are each a layer of an onion that has been sliced in half. This offers a further visual cue to the user and is also a practical means of control. An example of the alternating color trim and the combined control balls for three layers is shown close-up in Fig. 9, Bottom.   Once a group is formed, the user can drag all *onioned* control balls at once to raise and lower the combined layers as a single entity. The flattening of layers into 2D and the transparency of layers at a top height proceed as normal.

When grouping layers it is important to establish exactly how layers are to be shown with respect to other layers in the same group. To do this we need to consider the interaction of three different types of layers that may be present in a grouped layer namely 2D raster (image or array data based), 2D vector (line-based) and 3D layers (as will be discussed in sections 4.2-4.4). We first briefly list a number of cases that do *not* pose difficulties:

- Multiple vector layers, since vector lines do not occlude.

- Vector layers with a single raster layer, since vector layers can be placed slightly above the single raster layer, without occlusion.

- One or more 3D layers with vector layers and a single raster layer.

The above is true under the assumption that the vector and 3D layers are not overly dense. However, the case that always poses difficulties is using multiple raster layers in a single group, as they completely occlude each other. To overcome this problem, we introduce the notion of direct layer painting. Direct layer painting allows the user to reveal the data contained in one raster layer at the expense of all other raster layers in the same group. This effect is localized to wherever the user 'paints'.

Let us consider an example layer group that contains 3 raster layers. The three layers are added to the layer group in the following order: A then B then C. Initially, the raster layer that is completely visible is layer A, the first layer that was added to the group. Layers B and C are not visible. This default arrangement may not be sufficient as the user may wish to see certain portions of all three layers, A, B, and C, at the same time.

To tailor visibility within the group, the user can perform layer painting. The user first clicks on the name of the layer (within the layer legend) that he or she wishes to reveal portions of. We will assume this is layer B. The user then paints with the mouse directly onto the rendered area of the layer group. The areas that the user paints over will then only show information contained within layer B, thereby hiding data from layers A and C. Fig. 10 shows an example of this where 3 disjoint regions are painted for 3 raster layers within the group. Note that the regions have been color-

coded in this Fig. illustrate the concept; normally, the corresponding raster data from the 3 layers would be shown.

Fig. 10. Colour coded disjoint regions shown painted for 3 separate raster layers in a layer group.

The direct painting on a layer in 3D space raises technical issues. In particular, when the user clicks on a point in the display window, this does not specify a single point in 3D space because of the depth in the 3D scene. Instead, the mouse click casts an imaginary ray extending from the point on the display outwards to infinity. For our application, we must determine which point on this ray is the intersection point with the layer surface, which is the exact location the user intended to pick. As we use OpenGL as the underlying 3D graphics library we utilize the auxiliary function *gluUnproject* which performs the required reverse transformation of the projection from 3D space to 2D viewport space. Using this function we can trace a mouse click from a point on the screen to the target 3D point on the layer surface.

5. 3D Layers

In addition to the 2D vector and texture based layers previously described, we have augmented our system to include a number of 3D layers including point layers, landmark layers and chart layers.

5.2 Point Layers

Point layers are standard in traditional 2D GIS; in our system each point of data can be

represented by a 3D sphere (Fig. 11, top). The size of the sphere can represent one

aspatial aspect of the data points. For example, if each data point represents the

population of a town. Then the location of the sphere could represent the spatial data for

the town and the size of the sphere could then represent an aspatial data value for that
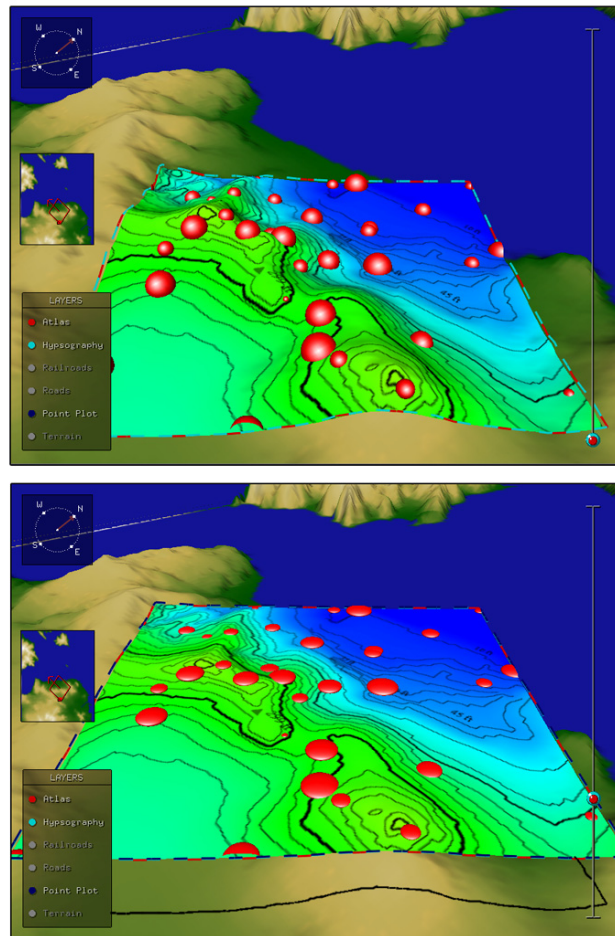
town such as the population density value.

Fig. 11. A partially transitioned point layer using sphere and disk symbols.

Each sphere is embedded in a layer and as the layer flattens, the spheres also flatten to form a 2D view (Fig. 11, bottom). The spheres also become transparent with the layer if the layer is raised sufficiently high. Also, it is of course possible to form multiple point-layers for different data content. Each point-layer would be assigned its own color and meta-data legend entry (to indicate the semantics of each point-layer).
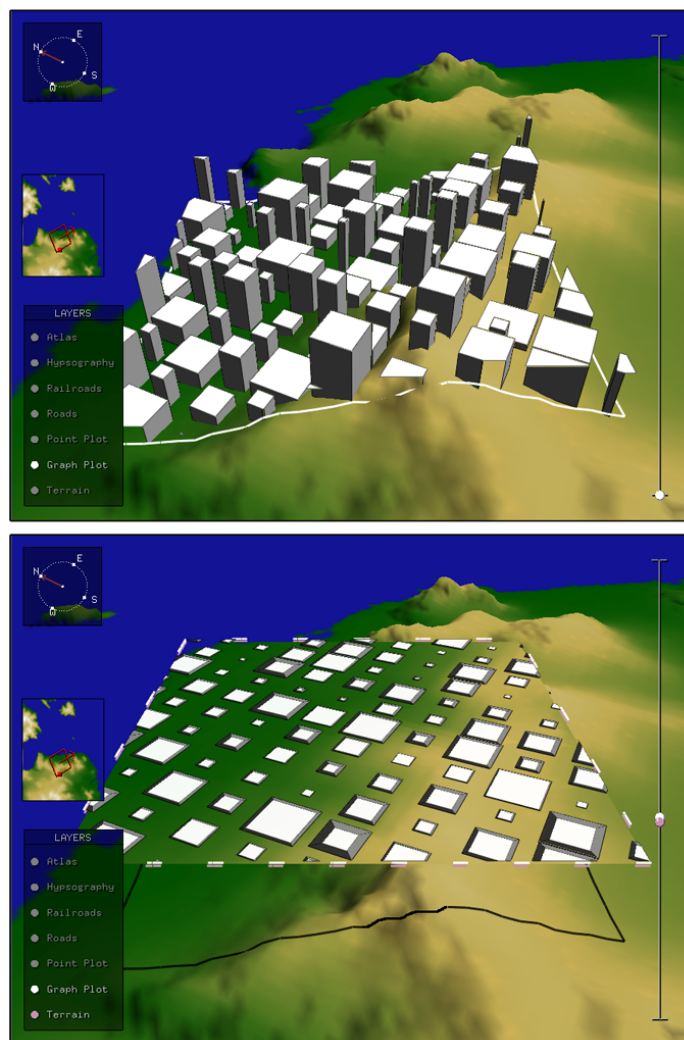

Fig. 12. A landmark layer a ground-level (left) and at the flattening level (right).

5.3 Landmark Layers

Integrating cityscapes in 3D GIS is not a new concept. Koomen et al. (2004) reported on the use of 3D GIS in urban development to visualise urban features and in particular urban volume. They found that 3D-visualisations have the advantage of identifying prominence of landmarks in a cityscape but are not as effective as 2D visualisations in providing an unambiguous, reproducible statistic of the urban exterior. We believe that there is a distinct advantage in having a flexible hybrid visualisation for systems that are design to aide urban planning and environmental impact studies.

In our visualisation we have integrated higher levels of detail and provided transitional landmark layers which represent major landmarks such as buildings in a cityscape (Fig. 12, Top). As a landmark layer is raised using the associated control ball, the buildings flatten gradually becoming 2D polygons on a 2D plane. During this flattening process the spatial information provided by the 3D view with respect to the relative building heights is lost. As a consequence, we have integrated visuals cues for the height of each building even when flat.

For this, we add a scaled edge trim for each building, indicative of the building's height when visualized in 2D (Fig. 12, Bottom). This scaled edge trim is implemented by scaling the top of the 3D building inwardly, proportional to the height of the building. In other words, the taller the building, the more of an 'edge' there is around the building when flat. This gives cue as to the building height even though the building is 2D.

The inward scaling factor, *f,* for the tops of the buildings is calculated as:

$$f(p,s) = p + (1-p)s \tag{1}$$

where:

$$p = \min(1, \max(0, 1 - h/H) + c) \tag{2}$$

$$s = 1 - (b/B) \qquad\qquad (3)$$

h	is current layer height,
H	is the height at which layers become flat,
b	is the current building's height,
B	is the height of the tallest building, and
c	is the minimum scale factor (default = 0.5).

Equation 3 computes a scaling factor, *s*, which scales the tops of taller buildings more than shorter buildings. Equation 2, limits this scaling so that the tops of the tallest buildings do not scale to a zero size. Equation 2 also takes into account the current layer height with respect to the maximum allowable layer height. In practice, this will mean that the higher the layer is raised, the more we need to scale all building tops inwardly. The *c* value in the equation 2 adjusts the maximum inward scaling for the top of the tallest building in the landmark layer when the layer is completely flat. All other buildings will scale by lesser amounts, proportionally.

Another key visual cue is that, although we flatten the buildings and scale the tops of buildings inwardly, we do not change the lighting properties of the buildings. In other words, we shade a building as if it was 3D, even though it has actually become flat. Technically, this means that we do not alter the surface *normals* on the building surfaces as it flattens. Normals are the outward facing direction of a polygon, being perpendicular to the surface. For a plane given by the equation $ax + by + cz = d$, the vector (a,b,c) is a normal. For a given polygon, the surface normal can be calculated as the vector cross product of two (non-parallel) edges of the polygon. By retaining the original normals computed for polygons that comprise a 3D building, the scaled edge trim retains the same shadowing as the building when it is not flattened.

For the purposes of illustration, the buildings used in this paper were algorithmically generated. However, it would be straightforward to extend our system

to import standard city file formats such as CityGML. The operational aspects of the

landmark layers themselves would not be significantly affected.

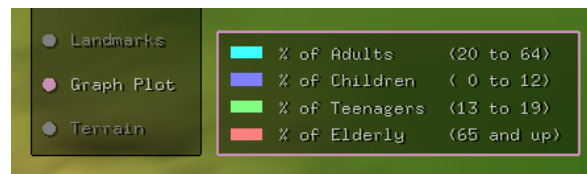Fig. 13. A transitioned chart layer employing a pie chart view.

Fig. 14. Auxiliary legend for pie chart classifications.

5.4 Chart Layers

Chart layers have also been integrated into the system and provide a way of visualizing aspatial data attributes using classifications. Our chart layers include 3D to 2D transitioning symbology as illustrated in Fig. 13. Classifications include both pie-size and color. The semantics of the classifications are available in an auxiliary attribute-classification legend. This legend is a pop-up legend that is visible when the chart layer is selected (Fig. 14).

One might argue that when the chart layer is viewed in 3D, it offers an immediate impression of the overall data with respect to the 3D spatial terrain data. In the 2D mode it is more visually precise and eliminates potential occlusions.

6 Camera Control and Zooming with the Layer System

We now describe how the virtual camera relates to the layer system. The layer system's position is fixed relative to the virtual camera meaning that the entire layering system moves with the viewer as they move across the landscape. The 2D layers are scaled and rotated synchronously with the 3D base layer view.

Translational movement of the camera over the base terrain is controlled with the arrow keys. When navigating through the terrain, the camera height is automatically adjusted based on terrain height under and in front of the camera.   The height is placed a certain distance *d* above the average local height of the terrain. The averaging of terrain elevation is done in a circular area around the camera position. The averaging results in the smoothing of the camera height adjustment, mitigating the effect of local bumps in the terrain and enables the user to avoid having to constantly adjust the camera height in order to remain above the terrain. Camera rotations pivot around the

centre of the layer system. This ensures that the user is viewing a consistent terrain location when rotating.

A further addition to our hybrid system is a seamless approach to camera zooming. This zooming feature allows the expansion and contraction of the terrain coverage, while maintaining a constant layer system interface. When zooming in and out, the layers, control balls and slider, legend and navigation tools all remain fixed in position and size relative to the camera. The only aspects that increase and decrease are the size of the terrain and the amount of terrain covered by the layers (Fig. 15).
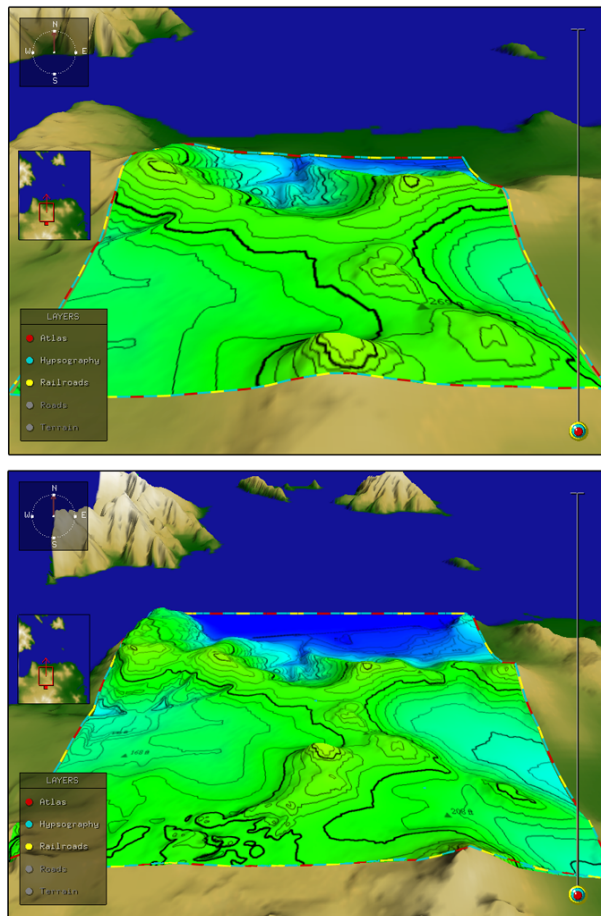

Fig. 15. Zooming in (top) and out (bottom) on a layer group.

In order to explain how this is implemented we must consider how each layer is rendered. Each layer's geometry is essentially the same as the base terrain itself. But, in addition to being (possibly) flattened, the layer is clipped on four sides with four clipping planes that are perpendicular to the layer. A clipping plane eliminates the rendering of all objects on one side of the plane. With four clipping planes we are able to eliminate all of the terrain geometry except for the square area that defines a layer. Each clipping plane is the same distance from the centre of the layer but along four opposing vectors. This ensures that we see a layer as a square area of the grid rather than the entire terrain.

In order to zoom, while maintaining a fixed layer size relative to the camera, we must simultaneously:

1) adjust the positions of the four clipping planes surrounding the layers with respect to the center of the layer system, and

2) move the camera backward or forwards along the camera's line of sight.

For example, if we are zooming-out to see a larger portion of the terrain, we must move all four clipping plane outwards with respect to the center of the layer system and simultaneously move the camera backwards along its line of sight. The relative rates of movement of both the camera and the four clipping planes must be precisely calculated in order to maintain the appearance of a constant layer system size.

7. Future Work

Since our system represents a novel approach to visualizing spatial data within a 3D GIS, there remain many opportunities for extending the system beyond its current form. One major extension will involve the addition GIS functionality such as advanced query

facilities. The integration of such querying functionality will make our system broadly applicable to a variety of tasks. The query results will form new layers and will exhibit 3D icons and labels. In order to provide such data query functionality we will need to integrate 3D spatial data into a database management system (DBMS). This will allow us to fuse spatial and aspatial data thus enabling interactive querying. Some researchers such as Li et al. (2003) are starting to address this issue and although a definitive solution is not yet available, steps forward are being made in this area of research.

To complement the querying facilities we also propose to add a data editing framework which would include dynamic rotations, translations, scale jumps and data selections. It is also our aim to undertake a usability study to confirm that a hybrid system such as ours is indeed advantageous to the GIS community. Other future work may include: level-of-detail visualization of both data and terrain geometry, the integration of time as a fourth dimension, and expanded symbology for existing layers. In summary, our approach has considerable scope for many future enhancements.

8. Conclusion

Our unique visualization system seamlessly integrates 2D and 3D views of the same data. The system allows the user to view the 2D data in direct relation to the 3D view within the same view. By combining traditional 2D GIS with a 3D view we are able to take advantage of both types of representations each with its own strengths.

We propose that by providing a 2D-3D transitional layer we can overcome both the self-occlusion and terrain-occlusion issues. Our layering system also offers a convenient means of handling multiple heterogeneous sets of aspatial data under user control. The system also allows the user to temporally set aside data that is not currently relevant. Further facilities which we have implemented include: landmark

layers, chart layers, 3D point layers, layer grouping, direct layer painting, the *onioning*

of layer control balls and a seamless zooming functionality. With the addition of these

innovative features, our system has taken significant steps towards a comprehensive

hybrid display.

Acknowledgements

References

Adelaide Model, Retrieved 10 October 2007 from
http://www.gisca.adelaide.edu.au/projects/am.html.

ATI ™, FireGL™ Graphics Cards, Retrieved 1 May 2007, from
http://ati.amd.com/products/workstation.html.

Brooks, S., & Whalley, J. (2005). A 2D/3D hybrid geographical information system. In
Proceedings of the ACM GRAPHITE Conference, Dunedin, 323-330.

Darken, R. P. & Silbert, J. L. (1993). A Toolset for Navigation in Virtual Environments.
In Proceedings of the ACM Symposium of User Interface Software and
Technology, 157-165.

ERDAS, Imagine Virtual GIS, Retrieved 1, May 2007, from
http://gi.leica-geosystems.com/.

ESRI ArcGIS. (n.d.). Retrieved May 1, 2007, from
http://www.esri.com/software/arcgis/.

Faust, N. (1995). Raster based GIS. GIS Development in North America, GIS World
Inc., July.

Hix, D. and Hartson, H.R. (1993). Developing User Interfaces: Ensuring usability
through product and process, Wiley, New York.

Hoppe, H. Progressive meshes. (1996). In Proceedings of ACM SIGGRAPH Computer
Graphics, 99-108.

Huang, B. and Lin, H. (1999). GeoVR: a web-based tool for virtual reality presentation
from 2D data. Computers and Geosciences, 25, 1167-1175.

Integraph, GeoMedia Terrain, Retrieved 1 May 2007, from
    http://www.intergraph.com/gterrain/.

Kapler, T. and Wright, W. (2005). Geo time information visualization. Information
    Visualization 4, 2, 136-146.

Köller, D., Lindstrom, P., Ribarsky, W., Hodges, L. F., Faust, N., & Turner, G. (1995).
    Virtual GIS: A real-time 3D geographic information system. In Proceedings of the
    IEEE Conference on Visualization '95, 94-100.

Koomen, E., Kaufholz, R., Rietveld, P. and Scholten, H. (2004). 3D-GIS and Urban
    Volume: Applying the Third Dimension in a Morphological Study of the
    Amsterdam Urban Landscape. In Proceedings of the 7th AGILE Conference on
    Geographical Information Science, 139-147.

Köeninger, A. and Bartel, S. (1998). 3D-GIS for Urban Purposes, GeoInformatica 2:1,
    79-103

Li, J., Jing, N. and Sun, M. (2001). Spatial database techniques oriented to visualization
    in 3D GIS. In Proceedings of the 2nd International Symposium on Digital Earth.

Lindstrom, P., Köller, D., Ribarsky, W., Hodges, L. F., Faust, N. and Turner, G.
    (1996). Real-time continuous level of details rendering of height fields. In
    Proceedings of ACM SIGGRAPH Computer Graphics, 109-118.

NVIDIA®, GeForce Graphics Cards, Retrieved 1 May 2007, from
    http://www.nvidia.com/object/geforce_family.html.

PCIGEOMATICS, PAMAP GIS Topographer, Retreived 1 May 2007, from
    http://www.pigeomatics.com.

Reddy, M., Leclerc, Y., Iverson, L. and Bletter, N. (1999). TerraVisionII: Visualizing
    massive terrain databases in VRML. IEEE Computer Graphics and Applications,
    19, 2, 30-38.

Rishe, N., Sun, Y., Chekmasov, M., Andriy, S., & Graham S. (2004). System
    architecture for 3D terrafly online GIS. In Proceedings of the IEEE Sixth
    International Symposium on Multimedia Software Engineering (MSE2004), 273-
    276.

Springmeyer, R. R., Blattner, M. M., Max, N. L., 1992. A characterization of the
    scientific data analysis process. In Proceedings IEEE Visualization, 235-242.

Stoter, J., & Zlatanova, S. (2003), 3D GIS where are we standing? Joint Workshop on
    Spatial, Temporal and Multi-Dimensional Data Modeling and Analysis, Quebec,
    Canada.

Troy, M., Möller, T., Atkins, M. S., & Kirkpatrick, A. E. (2004). Combining 2D and 3D Views for Orientation and Relative Position Tasks, In Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI 2004), 6(1), 73-80.

Tory, M., & Swindells, C. (2003). Comparing ExoVis, orientation icon, and in-place 3D visualization. In Proceedings of Graphics Interface, 57-64.

Tresens, M.A. and Kuester, F. (2003). Hybrid-reality: a collaborative environment for biomedical data exploration exploiting 2D and 3D correspondence. NSF Lake Tahoe Workshop on Collaborative Virtual Reality and Visualization, Lake Tahoe, USA.

van der Linden, J., Li, J., Lobb, R., Novins K., and Wuensche, B. (2004). strider: a simple and effective terrain navigation controller. Proceedings of IVCNZ '04, Akaroa, New Zealand, 95-100.

Verbee, E. G., van Maren, R., Germs, F.J and Kraak, M. (1999). Interation in virtual views – linking 3D GIS with VR. International Journal of Geographics Information Systems, 13, 4, 385-396.

Ware, C., Plumlee, M., Arsenault, R., Mayer, L. A., & Smith, S. (2001). GeoZui3d: data fusion for interpreting oceanographic data. In Proceedings of Oceans 2001, 3, 1960-1964.

Yao, R., Steemers, K. and Li, B. (2006). Sustainable Urban and Architectural Design. Chine Architecture and Building Press.

Yu, Q., Chen, C., Pan, Z. and Li, J. (2004). A GIS-based forest visual simulation system. In Proceedings of the Third International Conference on Image and Graphics, 410-413.

Zlatanova, S., Rahman, A., & Shi, W., (2004). Topological Models and frameworks for 3D spatial objects. Computers and Geosciences, 30, 419 – 428.

Zlatanova, S., Rahman, A., & Pilouk, M. (2002). Trends in 3D GIS development. Journal of Geospatial Engineering, 4(2), 1-10.

Biographies

Stephen Brooks is an Assistant Professor at Dalhousie University in Canada. Prior to this, he acquired a Ph.D. from the University of Cambridge in 2004 and a M.Sc. from the University of British Columbia in 2000.

Jacqueline L. Whalley is a Lecturer in Computer Science at the Auckland University of Technology in New Zealand. Jacqueline received her Ph.D. from the Australian National University and her M.Sc. from the University of Kent.