Vision, Modeling, and Visualization (2012)
M. Goesele, T. Grosch, B. Preim, H. Theisel, and K. Toennies (Eds.)

The Great Wall of Space-Time

C. Tominski and H.-J. Schulz

Institute for Computer Science, University of Rostock, Germany

Abstract

Understanding how data evolves in space and time is an essential task in many application domains. Despite the numerous visual methods that have been proposed to facilitate this task (e.g., showing the data on a map or plotting a time graph), the exploration of data with references to space and time still remains challenging. In this work, we present a novel concept for visualizing spatio-temporal data that refer to 2D geographical space and 1D linear time. The idea is to construct a non-planar slice – called the Great Wall of Space-Time – through the 3D (2D+1D) space-time continuum. Different visual representations can be projected onto the wall in order to display the data. We illustrate data visualizations based on color-coding and parallel coordinates. Compared to existing approaches, the wall has the advantage that it shows a closed path through space with no gaps between the information-bearing pixels on the screen. Hence, our novel visualization has the potential to be a useful addition to the user's toolbox of techniques for exploring the spatial and temporal evolution of data.

Categories and Subject Descriptors (according to ACM CCS): I.3.6 [Computer Graphics]: Miscellaneous—Visualization of spatio-temporal data

1. Introduction

Interactive exploration and visual analysis of spatio-temporal data are relevant in many application domains [AAD*10]. While there are many excellent solutions that focus either on the temporal aspects of data [AMST11] or on the spatial aspects of data [KO03], a major challenge is to understand the interplay of space, time, and spatio-temporal data values.

In the past, several researchers have developed techniques that integrate space, time, and data visually by embedding time-representing 3D glyphs into a space-representing 2D map display [TSWS05, TH10, FW10]. This kind of representation has the advantage that spatial and temporal aspects are shown within a single image. However, it is difficult to mentally link the information displayed in one 3D glyph to the information of another 3D glyph. This is due to the fact that the glyphs are separated spatially, that is, there is empty space between them. Figure 1 illustrates this problem: While the temporal evolution of the data is nicely visualized along the individual glyphs, the spatial evolution is difficult to extract, even for neighboring areas.

To alleviate this difficulty, we propose a novel technique that avoids gaps in the visual representation of the data. Our solution is to create a non-planar slice through 3D space-time, which we call the *Great Wall of Space-Time*. The wall is constructed based on the topological and geometrical properties of the spatial frame of reference. We provide interactive and automatic means to define the wall's path through space. Once erected, the wall can be used to visualize spatio-temporal data in different ways. Here we demonstrate visual representations based on color-coding individual bricks in the wall and based on projecting parallel coordinates onto the wall. Appropriate interaction techniques have been integrated to support users in exploring the data.

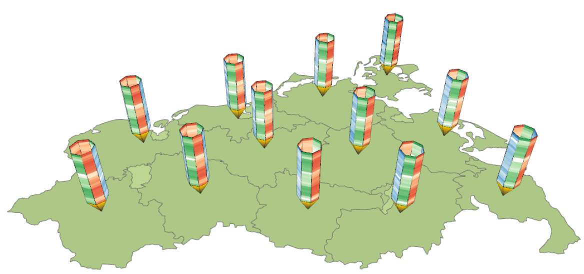


Figure 1: *Gaps between glyphs make it difficult for the user to relate the information displayed for the individual areas.*

2. Basics and Related Work

We consider spatio-temporal data that are defined as follows. The spatial dimension is composed of a set **A** of disjunct 2D geographical areas. The time dimension consists of a set **T** of discrete points in time, where we assume a linear time model. Space and time taken together define the domain in which spatio-temporal data have been collected. Our spatio-temporal data are stored as tuples of the form $(A, T, V_1, \ldots, V_n)$, where $A \in \mathbf{A}$ is a geographical area, $T \in \mathbf{T}$ is a point in time, and $V_i : 1 \leq i \leq n$ are the values measured at A and T. The challenge when visualizing such data is to integrate space, time, and data values.

In the visualization literature this challenge is dealt with in different ways. There are many techniques that focus either on the temporal aspect of the data [AMST11] or on the spatial aspect of the data [KO03]. In order to combine these techniques, one can use multiple coordinated views [WBWK00, Rob07], where multiple views show different aspects of the data, while the connection between space and time is realized via interactive brushing and linking [BMMS91]. One of the advantages of multiple view systems (e.g., VIS-STAMP [GCML06] or GAV [VHLÅJ12]) is their versatility and flexibility allowing the user to focus on different aspects of the data. On the other hand, the separation of the visualization into multiple views also puts some additional burden onto the user when it comes to understanding the interplay of spatial and temporal aspects.

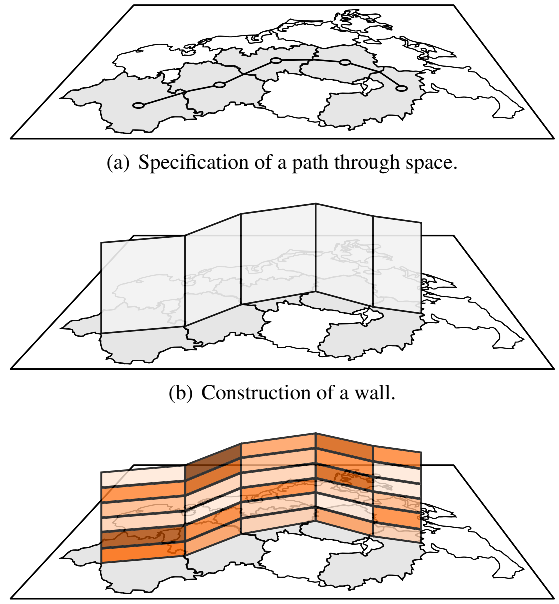
Therefore, researchers have long since been investigating visual representations for the direct integration of space and time. The most prominent example is the classic space-time cube [Häg70, Kra03]. Various approaches use the space-time cube as the underlying model for the visualization. One can distinguish between techniques that show collections of points in the space-time cube (e.g. [GAA04,KW05,TKB07]) and techniques that embed time-representing glyphs into the space-time cube (e.g., [TSWS05, TH10, FW10]).

Our work is concerned with the latter class of techniques. Figure 1 illustrates an example with glyphs embedded into a map display. Each glyph visualizes the time dependency of the data of its associated area. As illustrated in the figure, multiple variables can be color-coded along a glyph, which enables the user to compare temporal trends or patterns.

By placing multiple glyphs on the map, the spatial dependency of the data can be communicated. However, due to the spatial separation of the glyphs, it can be difficult for the user to understand how the data evolve in space.

3. Visualizing Spatio-Temporal Data as a Wall

Our goal is to provide a supplementary visualization that better supports the task of showing the spatial dependency, while still maintaining the visibility of the temporal dependency of the data. Next we first provide a brief overview of our approach and then explain it in detail.



(a) Specification of a path through space.

(b) Construction of a wall.

(c) Visualization of data on the wall.

Figure 2: *Basic idea of the Great Wall of Space-Time.*

3.1. Solution Overview

Given the fact that we deal with discrete geographical areas in 2D space and discrete points in 1D linear time, we can think of the data as a 3D space-time continuum or space-time cube [Häg70, Kra03]. The basic idea of our approach is to refrain from showing the data for all geographical areas, but instead to focus on one selected *slice* through the 3D space-time continuum at a time. Slice representations have been used since the early years of visualization research, in particular in the realm of volume visualization [NH90].

In contrast to classic slice-based visualization approaches, we do not consider a planar slice, but instead aim to create a meaningful topological path through space (see Figure 2(a)). From the topological path, we create a geometrical path (in the x/y plane) taking into account the geographic characteristics of the spatial frame of reference. The geometrical path is extruded vertically (along the z-axis) to form a wall-like 3D shape (see Figure 2(b)). This wall acts as a kind of canvas onto which we can project visual representations of the space- and time-dependent data. We map the dimension of time along the vertical extent of the wall and use color-coding to visualize individual data values (see Figure 2(c)). Alternative visual encodings are possible as well, for example the projection of parallel coordinates onto the wall.

This general approach of using a non-planar slice through space-time, constructing a wall, and projecting onto it visualizations of spatio-temporal data requires addressing the following aspects:

- **Meaningful topological path:** We must define what a "meaningful" topological path through space is. For this purpose, we consider the neighborhood graph induced by the partition of space into disjunct geographical areas.
- **Well-formed geometrical path:** The topological path has to be mapped to a geometrical representation. To this end, we consider the shapes of the geographical areas and make the geometrical path fit their spatial properties.
- **Visual mapping:** We need an appropriate visual mapping of data values onto the wall. Here we rely on well-accepted conventions from the visualization literature.
- **3D occlusion:** Because the 3D approach inherently leads to occlusion, we need mechanisms to deal with it. Our solution provides 3D navigation and visual adaptation tools to let the user look around and through the wall.
- **Interactive exploration:** As we address exploratory analysis scenarios, all steps must be interactively steerable by the user. Where appropriate, automatic methods are integrated to assist the user.

We will now describe our approach in more detail following the previously listed aspects.

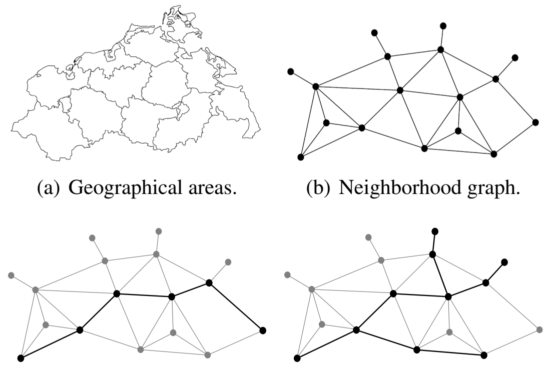
3.2. Topological Considerations

In order to construct the wall, we start with creating a topological path through space. This process is based the neighborhood graph $G = (\mathbf{A}, \mathbf{N})$. The set of vertices of this graph corresponds to the set of geographical areas $\mathbf{A}$. The set of edges $\mathbf{N}$ describes the neighborhood relationships of the areas: If the areas $A \in \mathbf{A}$ and $B \in \mathbf{A}$ are neighbors, there is an undirected edge $\{A, B\} \in \mathbf{N}$. Note that special cases such as islands or areas with holes (area genus > 0) can be handled by inserting dummy edges into the neighborhood graph.

We define our topological path via a subset of areas $\mathbf{A}' \subset \mathbf{A}$ such that the subgraph G' induced by $\mathbf{A}'$ is connected. The connectedness criterion is required to create a wall without gaps. Further we impose the constraint that G' be a *simple path* through G. This constraint guarantees that the wall does not self-intersect. To allow for more flexibility in the construction of the wall, we can loosen this constraint by considering an *acyclic subgraph* instead of a *simple path*. Then it is possible to create branching topological paths through space. The different variants are illustrated in Figure 3.

These topological considerations are the theoretical basis for the construction of the wall. In order to practically construct the wall, we need to provide means to specify which areas $\mathbf{A}'$ are to be part of the topological path. To this end, we developed interactive, semiautomatic, and automatic mechanisms that enable the user to design walls dynamically at runtime.

Interactive construction Full control is provided by the interactive mechanism. In this case the definition of a meaningful path is entirely based on input from the user, who specifies paths by successively selecting areas from the map. At all times, the mechanism offers only those areas for selection that lead to a well-defined continuation of the construction, as illustrated in Figure 4.



(a) Geographical areas. (b) Neighborhood graph.

(c) Simple path. (d) Acyclic subgraph.

Figure 3: *Topological aspects of the wall construction.*

Semi-automatic construction If the map contains many areas, selecting them interactively can be cumbersome. For such cases, we provide a semiautomatic construction mechanism. The only selections to be made by the user are the start area and the end area of the wall. The path in between the two selected areas is computed as the shortest path through the neighborhood graph. The computation can be based on the minimal number of areas in the path (see Figure 5(a)) or on minimal edge weights (e.g., geographic distance between area centers, see Figure 5(a)).

Automatic construction The aforementioned mechanisms operate on the topology of the geo-space, but they do not consider the spatio-temporal data. To further aid in the construction of the wall, we propose to automatically route the wall along trends in the data. Trends are often of interest when it comes to understanding how phenomena develop in space-time. In general, any spatio-temporal trend analysis can be utilized to drive the routing. A very simple and easy-to-implement approach is to apply a gradient descent. Starting with an initial area (e.g., area with maximum data value or interactively selected), we compare the area's data value at time T_i to the data values of the area's neighbors at time T_{i+1}. The neighbor with the biggest change is selected as the next area in the path. This procedure is repeated until no further areas can be added to the path. However, a simple gradient descend is prone to outliers and other data anomalies. Therefore, we recommend matching the choice of the trend analysis method to the characteristics of the data.

With the previously described methods the user can create abstract topological paths through space. In order to display a wall, we need to transform this abstract path into a geometrical representation.

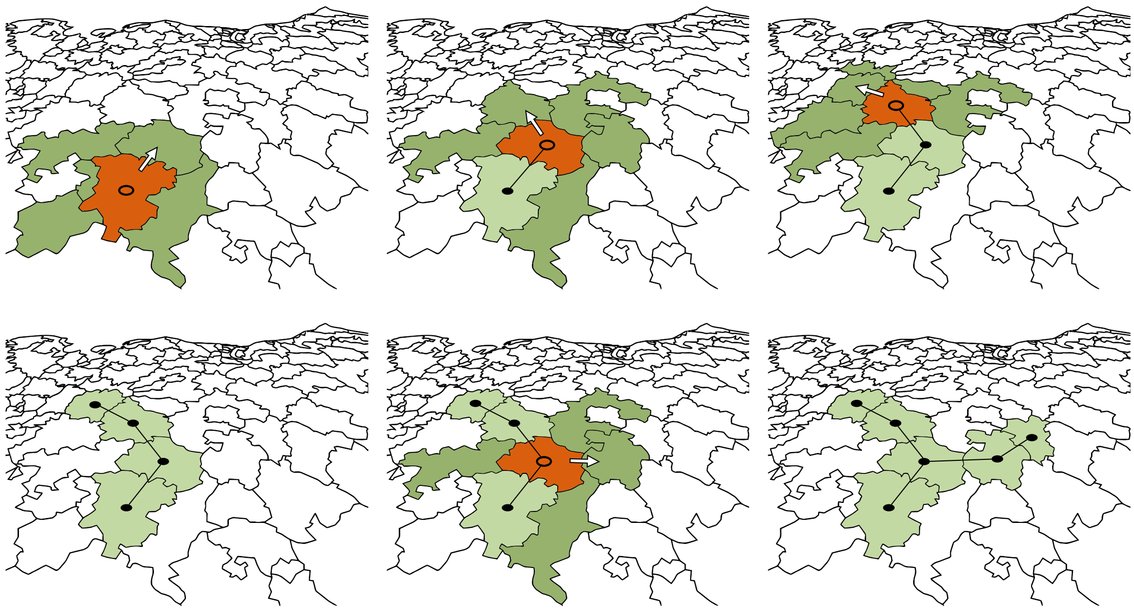
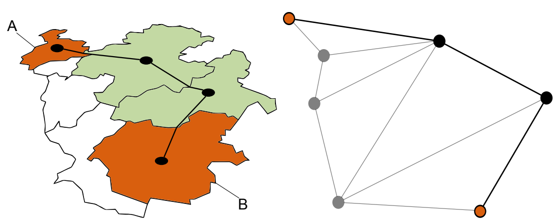
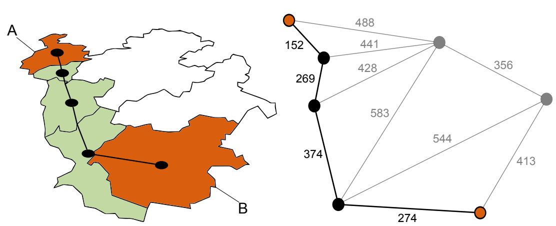


Figure 4: *Interactive specification of a branching topological path through space.*



(a) Shortest path based on fewest areas.



(b) Shortest path based on shortest distance.

Figure 5: *Semi-automatic path construction.*

3.3. Geometrical Considerations

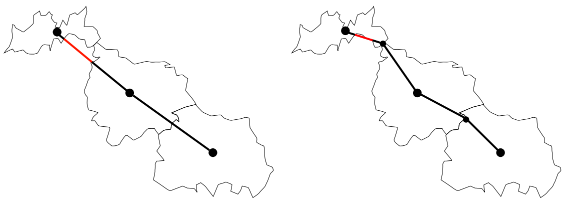
In the abstract topological path, each node is associated with an area of the spatial frame of reference. When lifting the abstract topological path to a geometrical path, we need to find a representation that corresponds to the geometrical features of these areas. Therefore, the construction of the geometrical path has to address the following two fundamental requirements:

- **Inclusion:** The geometrical path should stay within the areas that make up the topological path. In other words, the geometrical path should not cross areas that do not belong to the topological path.
- **Well-formedness** The geometrical path should be "well-shaped". Ideally, the geometrical path should be in the "center" of the area and be smooth with low curvature.

The difficulty is to fulfill these requirements given the arbitrary, often concave shape of geographical areas. We suggest two different strategies that vary in their computational complexity and in the degree to which they fulfill the stated requirements.

Simple Geometry Construction The most basic solution is to assign an anchor point to each area and connect the anchor points to form the geometrical path. The anchor points can be computed based on the center of mass or the center of the largest inscribed circle, or they can be set manually for extraordinarily complex areas. Connecting the anchor points with straight line segments results in a geometrical path as shown in Figure 6(a).

However, with this basic solution, parts of the geometrical path might be outside of the areas of the topological path (violation of the inclusion requirement). A simple way to alleviate, not to solve, this problem is to connect the anchor points via additional border points (see Figure 6(b)). Border points can be defined as the center point of the border shared by two areas. If the areas meet in a single point only, that point is trivially the border point.



(a) (b)

Figure 6: *Simple construction of geometrical path (a) without and (b) with border points. Red segments indicate violation of the inclusion requirement.*

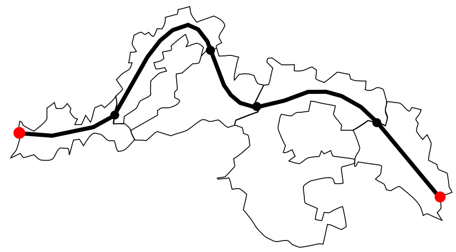


Figure 7: *Illustration of a geometrical path fulfilling the inclusion and well-formedness requirements.*

Using anchor points and border points, we can create geometrical paths that are acceptable in many cases, while the computational costs are kept low. However, we cannot ensure fulfillment of the requirements stated before.

Complex Geometry Construction To guarantee inclusion and well-formedness, methods need to be employed that are computationally more complex. We identified two methods that can be applied for this purpose.

One is to compute the skeletons and derive the medial axes of the areas (see Ogniewicz & Kübler [OK95]) participating in the topological path. By connecting the medial axes one can obtain a geometrical path that is in the "center" of the areas. However, the geometrical path is not smooth, making it necessary to apply an additional smoothing step to address the well-formedness criterion.

The second alternative is to utilize a technique to compute a shortest smooth path between any two points on the boundary of arbitrary polygonal shapes (see Abello & Gansner [AG98]). By successively computing such smooth paths between selected border points, one can construct a geometrical path with the desired features. This path is guaranteed to stay within the shapes of the areas, which meets our inclusion requirement, and the path is smooth, which meets our well-formedness requirement.

Both complex methods generate better geometrical paths than the simple method (see Figure 7). On the other hand, the computational as well as the implementation costs are much higher for the complex methods. For this reason, our prototype follows the simple method, which is sufficient for the purpose of demonstration.

Using the geometrical path, we can now construct a wall and visualize data on the wall as described next.

3.4. Visualizing Data on the Wall

The geometrical path through the 2D map space is the reference for the spatial dependency in the data. In order to account for the time aspect of the data, we need to define a reference that serves for the temporal dependency. For this purpose, the geometrical path is extruded along the vertical z-axis, which is used to encode time. The result of the extrusion is a wall-like geometrical object onto which visual representations of the data can be projected.

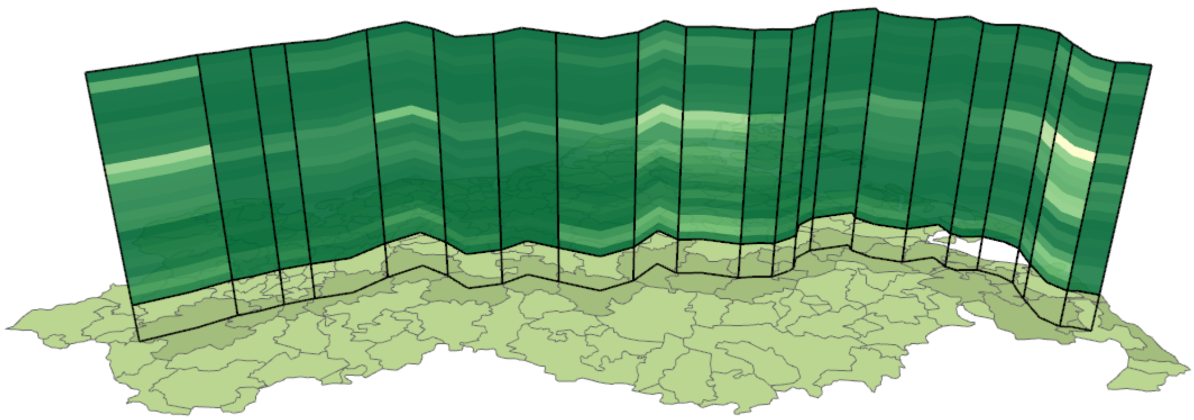
Our goal is to visualize the data such that there are no visual gaps between the areas along the wall. Therefore we decided for a color-coding of the wall. To visualize the data of the areas $\mathbf{A}' \subset \mathbf{A}$ along the wall for multiple points in time $\mathbf{T}' \subset \mathbf{T}$, we subdivide the wall according to $\mathbf{A}'$ and $\mathbf{T}'$. By this subdivision we obtain a wall that consists of individual bricks, each of which is associated with a unique area A and a unique point in time T. The bricks are then color-coded based on the data values stored for the individual areas and points in time.

For the color-coding, we rely on previous work on task-driven color-coding [TFS08]. Depending on the task at hand, the user can choose from predefined color scales from [BRT95] or [HB03]. On demand, the color scales can be automatically adapted to statistical properties of the data. Additional sliders enable the user to fine-tune the color coding.

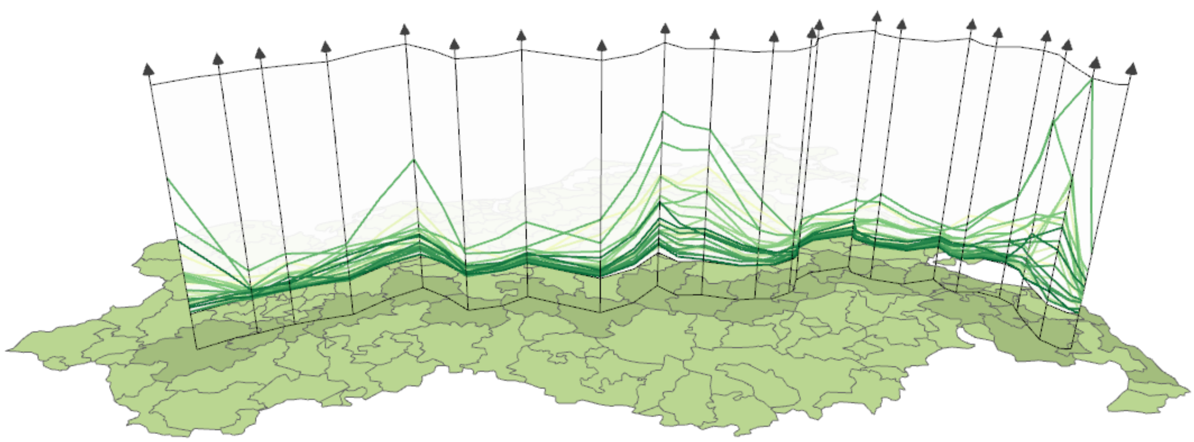
Figure 8(a) illustrates a color-coded wall with human health data. The path through space has been constructed manually and 24 months have been selected to be mapped along the vertical extent of the wall. Green bricks indicate a low number of cases of influenza, whereas yellowish bricks stand for high values. From the figure we can see that several areas have high numbers of cases in February and March as well as in October and November, which are the typical seasons for influenza. In the center of the figure we can also see two neighboring areas that show a quite similar pattern.

But colored bricks are not the only option for visualizing data. The wall can be considered a general projection surface that can show different visual representations with regard to the spatial and/or the temporal dependencies in the data.

One alternative, for example, is to project a parallel-coordinates-like visualization onto the wall. Figure 8(b) illustrates such a visual representation. For each area that the wall passes through, we place a vertical axis. Each axis represents the value range of the same user-selected data vari-



(a) Color-coded bricks.



(b) Parallel coordinates style.

Figure 8: *The Great Wall of Space-Time showing the number of cases of influenza per area and month.*

able (e.g., cases of influenza). For each point in time, we construct a polyline that connects the axes according to the underlying data values. In this case, the time-dependency of the data is no longer mapped along the vertical axes (which show the variables's value range). Therefore, we vary the saturation of the constructed polylines to indicate time. Of course, this works only with a limited number of time points.

The example in Figure 8(b) shows the same data as Figure 8(a). While the encoding of time is less obvious in the parallel-coordinates variant, other features can be seen better. Thanks to the encoding of values to positions on the axes, maximum values and outliers (e.g., fifth axis from left) can be discerned more precisely. Also the distribution of values is made more clear. As we can see some areas have consistently low values, while others show a wider span of values, with the extreme case show on the far right.

The presented visual encodings demonstrate that the wall design can be useful to visualize univariate spatio-temporal data. Beyond that, we see potential that a variety of goals can be achieved with the wall. One particular example is to compare the behavior of multiple variables. One option is to subdivide the bricks into sub-bricks, each of which being associated with a different data variable. In order to facilitate separability of variables, the sub-bricks should use distinct hues, while data values are encoded by varying saturation and/or brightness. Another option would be to adapt the parallel coordinates style to focus entirely on the spatial dependency of multiple variables (ignoring the temporal aspect). To this end, we show one polyline per variable, where separability of variables can be achieved by using a categorical color map [HB03]. A prerequisite for this option is that the value ranges be comparable.

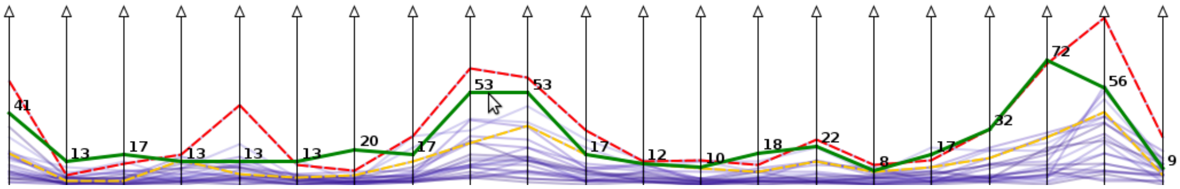


Figure 9: *A separate uniform and undistorted 2D representation complements the 3D wall display.*

3.5. Interaction

In order to explore the spatio-temporal data displayed on the wall, the user must be provided with adequate interaction techniques. The first issue we need to address is fluid and convenient 3D navigation to allow the user to look at the wall from any perspective. Our solution integrates free 3D fly-through, which supports an egocentric navigation in the view space, and orbit rotation combined with pan and zoom operations, by which we support object-centric adjustments of the perspective on the data.

The issue of 3D occlusion is addressed by providing the possibility to make the wall semi-transparent via alpha blending. However, the blending can have a negative impact on the perception of color-coded visualizations. Therefore, we added a second option to temporarily resolve occlusions. The user can raise and lower the visualization on the wall much like raising and dropping a curtain. Raising the visualization leaves free space at the bottom of the wall (see Figure 8), which is particularly useful to uncover the shapes of the areas contributing to the wall.

Because the wall's geometry depends on the shapes of the geographical areas, it can happen that some wall segments are rather small (then when the underlying area is small). Hence, the visual representation associated with small areas can be difficult to discern. To address this difficulty, users can apply an interactive cartographic fisheye lens, which temporarily distorts space to magnify smaller areas. The magnification affects both the map and the wall, which makes the data of smaller areas easier to explore.

To further address the difficulties with wall segments of varying sizes and 3D perspective distortion, the user can detach the visualized data from the wall and view them as a uniform and undistorted 2D representation. Coordinated highlighting is applied to maintain the connection between the data and the map. Figure 9 illustrates the 2D variant for the parallel coordinates style with the currently highlighted time point in green and its predecessor and successor in yellow and red, respectively.

On a more general level of interaction, our solution provides means to select the time range to be mapped to the wall and the variable(s) to be visualized. The interactive (and semi/automatic) tools to select areas for the wall were already mentioned earlier. Further interactive adjustment of visualization parameters is supported via dedicated controls. An example illustrating the controls for the task-driven color-coding is given in Figure 10. The user can select from a set of predefined color scales for identification and lookup tasks [AA06], switch between continuous and segmented color scales [BRT95], flip their orientation, apply statistical equalization methods [TFS08], and adjust the individual control points manually. A Box-Whisker plot provides an overview of the distribution of the data.

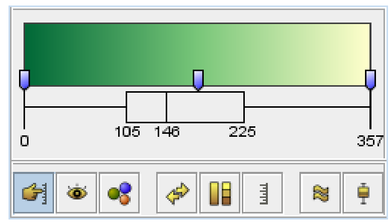


Figure 10: *User interface for adjusting the color-coding.*

4. Discussion and Conclusion

In summary, we presented a novel concept for visualizing spatio-temporal data. The visualization is based on the idea of creating a slice through space-time (interactively or automatically). The slicing corresponds to determining a path through the space dimension, whereas the time dimension is mapped along the vertical extent of the slice. The slice is used to construct a 3D wall onto which the visualization of the data is projected. We have shown that the visual mapping of spatio-temporal data can be done by means of color-coding and parallel coordinates. The visualization is complemented with interaction techniques to navigate in 3D, to deal with occlusion and small areas, and to adjust visualization parameters.

Our novel solution avoids gaps in the visual representation, which potentially has a positive impact on interpreting the spatial dependency of the data in addition to the temporal evolution. On the other hand, this comes at the price that the wall shows the spatial dependency only along the selected path. In this sense, our techniques makes a compromise within the conflicting priorities of showing space, time, and data in a single image. As a consequence, we understand our technique as a complementary tool that has to work together with other techniques to support all aspects of exploratory spatio-temporal data analysis.

This paper presented our initial ideas of using a wall for visualization. Although this general idea has already proved useful in a different application scenario (see [TSAA12]), there are still things to be explored in more detail.

In the future, one could investigate additional data characteristics that can be exploited to describe meaningful paths through space. Along with this comes the need to integrate analytical methods (e.g., spatial and time-series analysis methods) that extract such meaningful paths automatically. Further, the limitation to showing one selected path only has to be addressed. Using branching paths can alleviate this problem, but additional means are required allowing the user to compare multiple paths through space efficiently. Related to this is the questions of spatial and temporal resolution of the underlying data. If and how paths can be constructed across different levels of granularity remains an open questions that requires more research.

Finally, we need to conduct studies to evaluate the novel concept. Although initial informal feedback about the design was positive, more detailed evaluation is needed to identify strong aspects and weak spots of the concept.

Acknowledgments

We thank Mike Voigt for his help in implementing the concept and preparing many of the figures used in this work.

References

[AA06] ANDRIENKO N., ANDRIENKO G.: *Exploratory Analysis of Spatial and Temporal Data*. Springer, Berlin, 2006. 7

[AAD*10] ANDRIENKO G., ANDRIENKO N., DEMŠAR U., DRANSCH D., DYKES J., FABRIKANT S. I., JERN M., KRAAK M.-J., SCHUMANN H., TOMINSKI C.: Space, Time and Visual Analytics. *International Journal of Geographical Information Science 24*, 10 (2010), 1577–1600. 1

[AG98] ABELLO J., GANSNER E. R.: Short and Smooth Polygonal Paths. In *Proc. Latin American Symposium on Theoretical Informatics (LATIN)* (1998), vol. 1380 of *LNCS*, Springer, pp. 151–162. 5

[AMST11] AIGNER W., MIKSCH S., SCHUMANN H., TOMINSKI C.: *Visualization of Time-Oriented Data*. Springer, London, 2011. 1, 2

[BMMS91] BUJA A., MCDONALD J. A., MICHALAK J., STUETZLE W.: Interactive Data Visualization using Focusing and Linking. In *Proc. IEEE Visualization Conference (Vis)* (1991), IEEE Computer Society, pp. 156–163. 2

[BRT95] BERGMAN L. D., ROGOWITZ B. E., TREINISH L.: A Rule-Based Tool for Assisting Colormap Selection. In *Proc. IEEE Visualization Conference (Vis)* (1995), IEEE Computer Society, pp. 118–125. 5, 7

[FW10] FORLINES C., WITTENBURG K.: Wakame: Sense Making of Multi-Dimensional Spatial-Temporal Data. In *Proc. International Conference on Advanced Visual Interfaces (AVI)* (2010), ACM Press, pp. 33–40. 1, 2

[GAA04] GATALSKY P., ANDRIENKO N., ANDRIENKO G.: Interactive Analysis of Event Data Using Space-Time Cube. In *Proc. International Conference Information Visualisation (IV)* (2004), IEEE Computer Society, pp. 145–152. 2

[GCML06] GUO D., CHEN J., MACEACHREN A. M., LIAO K.: A Visualization System for Space-Time and Multivariate Patterns (VIS-STAMP). *IEEE Transactions on Visualization and Computer Graphics 12*, 6 (2006), 1461–1474. 2

[HB03] HARROWER M. A., BREWER C. A.: ColorBrewer.org: An Online Tool for Selecting Color Schemes for Maps. *The Cartographic Journal 40*, 1 (2003), 27–37. 5, 6

[Häg70] HÄGERSTRAND T.: What About People in Regional Science? *Papers of the Regional Science Association 24* (1970), 7–21. 2

[KO03] KRAAK M.-J., ORMELING F.: *Cartography: Visualization of Geospatial Data*, 2nd ed. Pearson Education, Harlow, 2003. 1, 2

[Kra03] KRAAK M.-J.: The Space-Time Cube Revisited from a Geovisualization Perspective. In *Proc. International Cartographic Conference (ICC)* (2003), The International Cartographic Association (ICA), pp. 1988–1995. 2

[KW05] KAPLER T., WRIGHT W.: GeoTime Information Visualization. *Information Visualization 4*, 2 (2005), 136–146. 2

[NH90] NIELSON G., HAMANN B.: Techniques for the Interactive Visualization of Volumetric Data. In *Proc. IEEE Visualization Conference (Vis)* (1990), IEEE Computer Society, pp. 45–50. 2

[OK95] OGNIEWICZ R., KÜBLER O.: Hierarchic Voronoi Skeletons. *Pattern Recognition 28*, 3 (1995), 343–359. 5

[Rob07] ROBERTS J.: State of the Art: Coordinated Multiple Views in Exploratory Visualization. In *Proc. International Conference on Coordinated and Multiple Views in Exploratory Visualization (CMV)* (2007), IEEE Computer Society, pp. 61–71. 2

[TFS08] TOMINSKI C., FUCHS G., SCHUMANN H.: Task-Driven Color Coding. In *Proc. International Conference Information Visualisation (IV)* (2008), pp. 373–380. 5, 7

[TH10] THAKUR S., HANSON A. J.: A 3D Visualization of Multiple Time Series on Maps. In *Proc. International Conference Information Visualisation (IV)* (2010), IEEE Computer Society, pp. 336–343. 1, 2

[TKB07] TURDUKULOV U. D., KRAAK M.-J., BLOK C. A.: Designing a Visual Environment for Exploration of Time Series of Remote Sensing Data: In Search for Convective Clouds. *Computers & Graphics 31*, 3 (2007), 370–379. 2

[TSAA12] TOMINSKI C., SCHUMANN H., ANDRIENKO G., ANDRIENKO N.: Stacking-Based Visualization of Trajectory Attribute Data. *IEEE Transactions on Visualization and Computer Graphics 18*, 12 (2012). to appear. 8

[TSWS05] TOMINSKI C., SCHULZE-WOLLGAST P., SCHUMANN H.: 3D Information Visualization for Time Dependent Data on Maps. In *Proc. International Conference Information Visualisation (IV)* (2005), pp. 175–181. 1, 2

[VHLÅJ12] VAN HO Q., LUNDBLAD P., ÅSTRÖM T., JERN M.: A Web-Enabled Visualization Toolkit for Geovisual Analytics. *Information Visualization 11*, 1 (2012), 22–42. 2

[WBWK00] WANG BALDONADO M. Q., WOODRUFF A., KUCHINSKY A.: Guidelines for Using Multiple Views in Information Visualization. In *Proc. Working Conference on Advanced Visual Interfaces (AVI)* (2000), ACM Press, pp. 110–119. 2