Visualization framework for the integration and exploration of heterogeneous geospatial data

Alexander Greß and Reinhard Klein

Institute of Computer Science II, Universität Bonn, Germany

Abstract

This paper presents an interactive visualization framework for heterogeneous geospatial data developed in context of an interdisciplinary research project that aims at the risk analysis of sea-dumped chemical weapons in the Baltic Sea. In the focus of the analysis are geophysical, hydrographical, geochemical, and biological data acquired on research cruises as well as data produced by toxic compound migration and bioaccumulation modeling. These different types of data to be visualized are represented as height fields, 2D vector maps, seismic profiles, and time-dependent scalar and vector fields. In general, these datasets are given at largely varying resolutions and geospatial extents, which makes their integration into one visualization especially challenging and requires efficient level-of-detail techniques. Furthermore, special care is taken on the appropriate integration and efficient 3D visualization of all different types of data at the same time. Several examples demonstrate the effectiveness of the resulting visualizations for collaborative analysis of the data.

Categories and Subject Descriptors (according to ACM CCS): I.3.3 [Computer Graphics]: Applications

1. Introduction

Interactive information visualization plays an important role in geosciences, especially in context of data acquisition and data analysis. For this reason, there exist a lot of interactive tools for visualization and analysis of specific geospatial data and specific visualization requirements. While these tools are effective for geospatial data analysis tasks in several different disciplines, there is a need for an interdisciplinary visualization and analysis framework that allows researchers and users from different fields to collaborate and explore their data in a common framework. This leads to additional requirements:

One of them is the need to visualize and explore heterogeneous data from different disciplines simultaneously to be able to draw conclusions based on all available data. An important issue in this context is an appropriate integration of data that are used on different scales in the different communities, such as bathymetry data that are in general used only at low resolutions for modeling of hydrophysical processes and ecosystem modeling, but are needed at high resolution in context of the inspection of geophysical properties.

Moreover, the geographic extents of the considered datasets in the different communities vary largely, for instance data gathered on surveys usually focus on narrow and restricted areas, while datasets used within computational models such as ecosystem models have usually a much larger extent to be able to draw conclusions for larger geographic areas, for instance to assess potential threats to the environment.

Conventional GIS applications (geographic information systems) usually offer good possibilities for viewing and exploring geospatial data in 2D, but often provide only limited 3D functionality whose quality and efficiency is not sufficient for interactive exploration of the data on varying scales. However, efficient 3D visualization is important for a collaborative visualization framework, since for an appropriate integration of different geospatial datasets, their respective geographic locations in all three dimensions have to be taken into account, and since usually at least some of the geospatial datasets to be integrated are by itself of three-dimensional nature.

For this reason, we developed an innovative framework for interdisciplinary visualization tasks with special focus on the appropriate integration and efficient 3D visualization of heterogeneous data of varying extents and varying resolu-

tion. By incorporating level-of-detail techniques, continuous zooming and fast navigation in 3D is made possible.

The use of 3D visualization helps also for representing the data in such a way that not only specialists of the respective discipline are able to interpret them, but to make the visualized data also intuitively understandable for other users, which is important for enabling effective collaboration as well as for the presentation of interdisciplinary research results.

2. Challenges and related work

Related work. First of all, there are many classic GIS applications (e.g. ArcView, Global Mapper), whose primary focus is on data analysis and processing as well as 2D visualization. Therefore, they usually have only restricted 3D functionality and no focus on real-time rendering of complex 3D datasets.

Additionally, in the recent years several 3D geospatial viewer application have been released (e.g. Google Earth, Microsoft Virtual Earth), some of them combined with basic GIS application functionality (e.g. ArcGIS Explorer, NASA World Wind). These applications implement efficient level-of-detail rendering, but while their range of detail levels is very high with respect to texture data, it is usually quite limited with respect to geometry, especially in case of non-terrain data.

Besides, there are many classic applications for visualization of scientific data, some of them with very wide applicability (e.g. Vis5d+, IDV) while others focus on specific application areas such as mining (e.g. Fledermaus [MPG*00], MVS, RockWorks). Some of these visualization tools can be integrated into GIS applications (e.g. EVS for ArcView) or can interface with them. Furthermore, several general frameworks exist that can be used to build specific visualization solutions via visual programming and other development techniques (e.g. OpenDX [TBF00], VTK [SML06], AVS). All these visualization applications and frameworks provide 3D display functionality and several of them can also handle large datasets efficiently, but as their primary focus is on data analysis and processing, visualization of very complex heterogeneous datasets is usually not possible in real-time or only with considerably reduced visual quality.

Challenges. Since our aim is to create an integrated framework for the efficient and intuitive inspection and presentation of heterogeneous geospatial data, which includes classic 2D maps and related GIS datasets as well as scientific data of three-dimensional shape, an important challenge was to find an approach suitable for real-time multi-resolution 3D visualization of such heterogeneous datasets that is capable of handling very complex datasets and the combination of datasets at very different scales.

For this reason, we decided to base our visualization framework not on a classic GIS or scientific visualization framework, but on a real-time rendering engine that enables efficient continuous level-of-detail rendering (SCARPED [WMD*04]). To that extent the engine we use has similar capabilities as the 3D geospatial viewer applications mentioned above, but it can even visualize a wider range of geometry detail levels efficiently than most alternative continuous level-of-detail rendering approaches. Although this engine was originally developed as terrain rendering approach only, its level-of-detail concept has recently been extended to include other types of geometries and to enable semantic interaction (see e.g. [WK07]). The approach makes use of well-designed multi-threaded caching and efficient GPU-based rendering techniques and is capable of streaming geometry and texture data from large out-of-core repositories from disk or via network. The engine has also been used for several commercial applications such as interactive 3D skiing maps with very high resolution geometry data.

One important issue for our visualization framework is to add functionality known from the context of classic GIS and visualization applications while still retaining the high rendering efficiency in context of complex and heterogeneous datasets. Furthermore, the exchange of data with other applications is especially important for a framework that aims at the integration of a wide range of heterogeneous datasets; thus our application has to provide sufficient flexibility with respect to data and file formats. Fortunately, we can make use of an open-source library that is often employed in this context (GDAL [OSG]) as foundation for adding basic GIS functionality and assuring compatibility to major GIS applications. However, it is important to note that the integration of GIS-typical functionality such as mapping of 2D shapes onto the 3D base geometry (terrain) has to take the multi-resolution representation of the base geometry into account.

Since we want to allow all input data to be given in an arbitrary geographic reference system, all geometry must be transformed precisely and efficiently to the coordinate space used for rendering; and since we want to support also 2D visualization using a geographic map projection of the user's choice (which is needed for efficient preparation of precise 2D maps as well as for real-time navigation in 2D), an additional challenge was to perform the required non-linear transformations as efficiently as possible.

Furthermore, our application should be able to visualize other types of scientific datasets. While several open-source frameworks are available for this purpose, virtually all of them do not focus on efficient GPU-based rendering of complex datasets, which is however important in our context of combining multiple heterogeneous data sources. For our specific visualization tasks, we therefore preferred other approaches (as detailed in Section 4) that are either based on our multi-resolution framework (for those areas where huge amounts of data are collected and visualized) or that make use of specific efficient GPU-based techniques for the

respective visualization task. This way we enable efficient real-time rendering of complex scientific datasets, which is important in our setting.

For the purpose of combining data produced from several different visualization tools, the compatibility in terms of data formats is of importance, which turned out to be not trivial because of the large amount of different, incompatible data and file formats used by scientific visualization applications. In particular, the geometry of grid-based datasets is commonly given only implicitly in different coordinate spaces (which are in case of modeled data often inherent to the underlying simulation). For geospatial visualization of these data, coordinates in the geographic space have to be reproduced, which in general requires some knowledge of the underlying model. To ensure compatibility to many different visualization tools, we integrated an existing framework for unification of these data models (Unidata's CDM [UCA]) and extended it appropriately to enable real-time inspection of scientific data produced by other applications in a true geospatial context.

Thus in summary, we based our visualization framework on several different building blocks in order to combine the advantages of approaches from different scopes while enhancing these approaches to our specific requirements wherever necessary. After describing what these requirements are in our case (Section 3), we will detail our solutions to the challenges sketched in this section (Sections 4 and 5).

3. Specific requirements

Our visualization framework was developed in context of the interdisciplinary research project "Modelling of Ecological Risks related to Sea-Dumped Chemical Weapons" (MERCW) funded by the European Commission. The MERCW project focuses on the study of chemical munition dumpsites in the Baltic Sea. After the First and Second World War, large quantities of chemical warfare were dumped in these areas, forming a potential threat to the marine environment and the population of the coastal zones. The objective of the MERCW project is to assess the ecological risks related to the dumped warfare through site investigations and modeling.

In context of this project, many different types of data and information were gathered and produced. This includes geophysical, hydrographical, geochemical, and biological data acquired on research cruises as well as data produced by toxic compound migration and bioaccumulation modeling. This implies several specific requirements and poses additional challenges to our visualization framework as detailed in the following.

3.1. Topographic and bathymetric data

In context of the MERCW project, high-resolution bathymetric measurements have been acquired on several research cruises. One network of survey lines covers the whole area of the primary dumpsite next to Bornholm. The lines of this network have a spacing of approximately 500 m and 1000 m in latitude and longitude direction, respectively. The sample points along each line are less than a meter apart. In contrast, all publicly available bathymetry datasets from this region have grid spacings not finer than 500 m to 2000 m. Due to the much higher resolution of the bathymetric measurements from the project, at least along the survey lines, some interesting features are clearly recognizable in this bathymetric data (e.g. shipwrecks), which are not observable in lower resolution data. Because of this, it seemed to be very valuable to integrate these high-resolution data in a digital terrain model of the Baltic Sea bathymetry.

While the efficient storage and visualization of the digital terrain in form of a multi-resolution model is handled by the level-of-detail rendering engine that we use, the task to *compose* a combined terrain model from the whole Baltic Sea region based on the available data sources in a suitable way was very challenging, since although there are several bathymetric datasets from the considered region available, their data quality differs locally based on the type and amount of measurements integrated into the respective dataset; and none of them could be identified to be clearly superior for the overall area. Furthermore, some of these datasets contain areas for which no utilizable bathymetry data is available. We call them *no-data areas*. This includes areas explicitly masked out as well as areas of invalid data that have been detected by visual inspection, by outlier detection or by comparison with other datasets. For some of the considered datasets, the no-data areas have non-regularly shaped boundaries. Thus an approach had to be devised that allows to combine these datasets in a way that the best available and most plausible values are used at every point and that still no seams or other artifacts are visible at the boundaries between the datasets, also in case of combining the high-resolution and low-resolution datasets. We will describe our approach to this challenge in Section 5.

3.2. Sample data and vector maps

One type of data acquired during surveys is data measured at irregularly distributed locations, such as samples taken from certain locations of interest or positions of shipwrecks. The visualization should provide an overview of the locations of these measurements and should on request also show other information associated with these locations (metadata). Another type of data that has to be included in the visualization are 2D shapes, e.g. the ship tracks, or other data that can be represented by 2D vector maps.

3.3. Seismic profiles

In addition to bathymetric data, also seismic and other geophysical data has been collected on the research cruises of
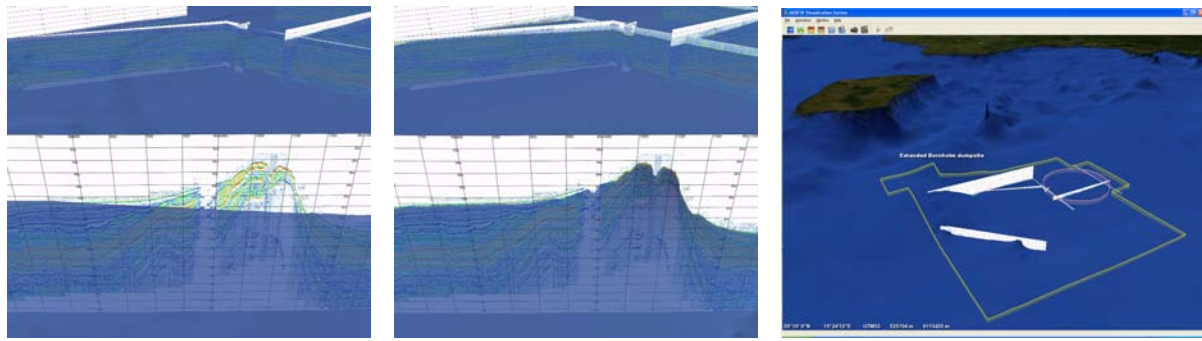
Figure 1: *3D visualization of seismic profiles in combination with the digital terrain model (drawn semi-transparently in the first two figures). The leftmost figure shows the terrain model without having integrated the high-resolution bathymetry, while the other figures show the final model. It can be seen in the center figure that the uppermost characteristic line in the profile images coincides with the terrain model surface for all profiles, while in the left figure, the terrain surface does not coincide with this line at all. The figure on the right, showing an overview of the dumpsite area in front of the islands Bornholm and Christiansø, additionally demonstrates that the high-resolution data was blended smoothly into the low-resolution bathymetry.*

the MERCW project. Seismic profiles have been acquired along the survey lines of several networks in the dumpsite area close to Bornholm. The 2D profile images generated for each network line reveal structural information for a certain depth range below the seabed. While in general the spacing between the network lines is too large to generate meaningful volumetric datasets from the seismic data by means of interpolation, it is more desirable to visualize the data in 3D space at exactly those places where they were measured, i.e. in form of vertically extruded network lines textured with the 2D profile images. Because of the high measurement resolution along the lines and in depth, the amount of profile image data to visualize is considerably high. Also the amount of corresponding geometry for these survey lines is non-trivial, since the ship tracks do not correspond to exact straight lines. Another challenge when visualizing the seismic profiles together with the terrain model is that the height of the seabed as represented by the terrain model should coincide with the corresponding information recognizable in the profile images, see Figure 1. (We will come back to that issue in Section 5.)

3.4. Multi-dimensional scalar fields

One important task in the project related to the assessment of potential risks was the modeling of predicted environmental concentrations (PEC) of chemical warfare agents (CWA) and thus also the visualization of the results of this modeling. This includes datasets of different shape and dimensionality (areal and volumetric ones, datasets defined on a radial domain and those defined on a rectilinear domain, etc.). Some of the datasets to visualize are time-dependent, i.e. they represent modeling results for multiple points in time within a considered time period. The common property of all datasets is that they can be characterized as multidimensional grid datasets. A special challenge in our context was to develop a solution that is as general applicable as possible in the sense

that a large amount of different grid types and underlying coordinate systems is supported.

4. Our visualization framework in detail

Our real-time visualization system is capable of displaying sample data and vector maps, seismic profiles, multi-dimensional scalar fields, and hydrodynamics concurrently on top of a detailed multi-resolution terrain model, which is described in the following. Furthermore, several GIS-like interaction and exploration facilities are provided, which are outside the scope of this paper.

4.1. Visualization of topographic and bathymetric data

As mentioned in Section 2, the visualization of bathymetric and topographic data in our system is realized based on the approach presented in [WMD*04], which is a very efficient level-of-detail rendering approach that reproduces the terrain dataset with guaranteed precision and guaranteed frame rate.

One of the aforementioned challenges in this context was the efficient run-time transformation of the geometry from the coordinate spaces in which the bathymetric and topographic datasets were given to the respective coordinate space used for rendering (which can be a Cartesian 3D space, a geodetic 2D space, or some projective 2D space, dependent on the user's choice). For efficiency, we perform the exact non-linear transformation at run-time only for a well-defined subset of the geometry points, and determine the transformed positions of the other points via bilinear or bicubic interpolation from neighboring transformed points on the GPU. The denseness of the subset of precisely transformed points is chosen as such that the positions of the remaining points can be determined from them with sufficient screen space precision.
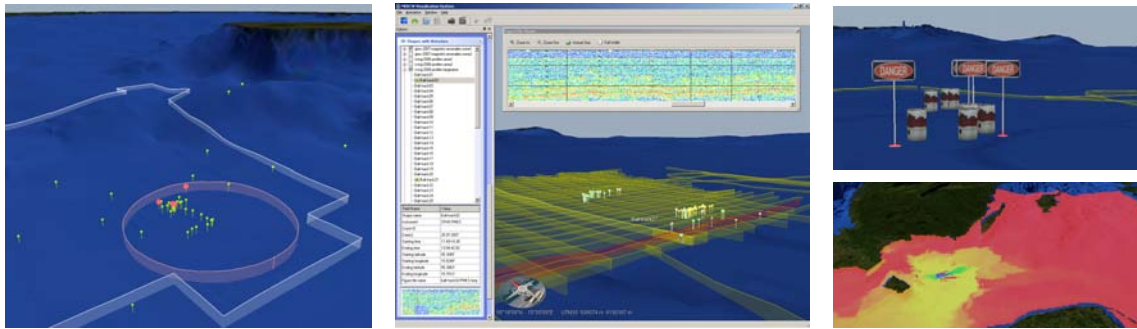
Figure 2: *Left: Visualization of sample locations and boundary lines. Center: Sample locations and network lines with corresponding metadata. Right top: Using 3D icons to visualize sample points. Right bottom: Visualization of a 3D scalar field.*

An additional challenge in our specific context was to improve the depth perception of the viewer when looking at the relatively shallow Baltic Sea bathymetry. Beside through the use of dynamic lighting and the support of stereoscopic rendering, we achieved this via the following techniques:

- First, the user has the ability to adjust the vertical scaling of the terrain model for visualization via separate scaling factors for bathymetry and land topography. The scaling is performed at run-time on the GPU during the visualization of the terrain model (together with the coordinate space transformation described previously). Of course, other types of data visualized on top of the terrain have to be scaled accordingly.
- Second, the visualization system allows overlaying isolines onto the terrain surface (for 2D as well as 3D visualizations), similar to the traditional visualization of these data in 2D GIS applications. Contour lines with fixed line width in screen space are determined in run-time on the GPU (using derivative functions in the fragment shader). The distance between these lines can be interactively adjusted by the user.

4.2. Visualization of sample data and vector maps

For visualizing sample data and 2D vector maps, we integrated three alternative techniques into our system:

- One option is to visualize the point and line primitives at a certain height in 3D space. To make these primitives visible at all scales of the visualization, we render them with a fixed (minimum) screen space size. In addition, to make their exact location better recognizable in 3D space, we connect point shapes by a thin line with the seabed and likewise line shapes by a thin polygon, see Figure 2.
- Another option is to project the primitives onto the terrain. In this case, it is important to avoid occlusion and z-fighting artifacts, which is especially challenging with multi-resolution terrain. Geometry-based rendering approaches are thus not suited here. Instead, we use a stencil-based rendering technique based on [SK07] for this type of visualization.

- A further alternative in case of sample data is the display of 3D icons: Instead of points or 2D point sprites, we allow any regular 3D object to be used for representing certain locations, see Figure 2 right top.

4.3. Visualization of seismic profiles

We visualize seismic profiles as textured, vertically extruded curves on top of the 3D terrain, see Figure 1 and Figure 3 right, similar to the curtain plots used in [WPA*01]. However, in our setting we have to cope with a huge amount of profile data, as mentioned in Section 3.3. Therefore, to enable efficient real-time rendering of these profile datasets in 3D space, it was inevitable to base the visualization of the profile data on a continuous level-of-detail rendering technique similar to the one used for visualizing the terrain.

In detail, we use a quad-tree representation for handling the geometry and textures of the seismic profiles as well as appropriate caching strategies. The profile images are stored in JPEG or PNG format on disk and are decoded on demand simultaneously to the visualization in a separate application thread. By using a progressive rendering technique, a fixed worst-case frame rate for the visualization of the profiles is guaranteed. This results in a high-quality real-time 3D visualization of the seismic profiles, which is, to our knowledge, not found in any other existing visualization tool.

4.4. Visualization of multi-dimensional scalar fields

Our system can visualize 2D – 4D scalar fields (see e.g. Figure 2 right bottom and Figure 3 center). By using a unified storage model for many different grid types and by supporting many different grid coordinate systems including those specific to the oceanographic domain (such as the ocean sigma coordinate system), we ensured compatibility to data generated from many other existing applications. In general, we use color coding with a user-customizable color transfer function for the visualization of the scalar fields. In case of volumetric datasets, we usually want to inspect horizontal grid layers (e.g. so-called sigma layers) separately. Note that dependent on the grid coordinate systems, such layers can

have complex, non-planar geometry in the coordinate space used for rendering. If a given grid dataset contains no-data areas, we encode the location of these areas in an alpha map, and use this map to exclude the no-data areas from visualization (by means of alpha blending). To allow a smooth animation playback when a time-dependent grid dataset is shown, the data is interpolated linearly between the given time frames. This is realized efficiently using a 3D texture for each grid layer, where the third texture coordinate dimension represents the time axis.

4.5. Visualization of hydrodynamics

Another challenge was to visualize hydrodynamics aspects efficiently using particle systems. The first use of particle systems in our application is for the purpose of visualizing sea currents. In that case, massless particles are assumed, whose velocities can be derived from given current velocities. Vector fields with modeled current velocity data can be obtained from an ocean model simulation. We perform particle simulations based on such vector fields on the GPU following the approach of Krüger et al. [KKKW05], which allows to simulate and render a few millions of particles in real-time. The approach was adapted to incorporate the non-linear transformation between the distinct coordinate spaces used for simulation and rendering (cf. Section 4.1). During the real-time simulation, the user can specify the positions or areas where particles are to be emitted interactively.

The second use of particle systems is to visualize the motion that real particles would undergo when emitted into the sea, e.g. to study the spreading of toxic compounds. Here, a much more complex dynamics model has to be applied, which is in general not suited for real-time evaluation. In such a case, we load the particle simulation result from disk and perform only transformation and rendering in real-time.

In both cases, the particles can be visualized in various ways, e.g. as spheres or as small arrows indicating the local directions of the flow (by utilizing the efficient rendering technique proposed by [KKKW05]) or by rendering their trajectories as lines, see Figure 3.

5. Composition of the terrain model

As pointed out in Section 3.1, for visualizing topographic and bathymetric data in our system, the task to construct a complete multi-resolution dataset based on several potentially incomplete datasets of different resolutions had to be solved. In particular, arbitrarily shaped areas from several datasets should be combined into a common dataset given only the information what dataset represents which region best. Also datasets that include no-data areas should be integrated properly into the final terrain model, i.e. without blending in data from within these areas. Another requirement is that the combination of multiple datasets should be totally seamless. Because of the way the input datasets have

been constructed based on different types and amounts of bathymetric measurements and different techniques for integrating them into a bathymetry grid and finally also because of the different resolutions (especially when trying to combine the high-resolution survey line bathymetry with the bathymetric grids of much lower resolution), the bathymetry for a certain location can differ significantly between multiple datasets. Simply replacing parts of one dataset by the corresponding regions from another dataset would produce seams clearly recognizable in the visualization of the resulting digital terrain model. Instead, a technique is needed that blends smoothly between the different datasets, i.e. the transition should be not noticeable in the resulting terrain model visualization. And at the same time, the combined dataset should adhere as closely to the input datasets as possible, even in the regions of transition between different datasets.

By extending recent methods used in similar context, we devised two techniques that fulfill all these requirements (and also deal with the issue mentioned in Section 3.3). While in principle both techniques can be used to compose multiple incomplete bathymetric datasets, the first one (described in Section 5.1) is especially well suited for combining multiple datasets with small to medium-sized no-data areas, which is characteristic for all the publicly available bathymetric datasets that we considered. For such datasets, this technique can guarantee smooth transitions between the datasets while making sure that no invalid assumptions about the global characteristics of the bathymetry are made.

In contrast, the compositing technique described in Section 5.2 is also suited for combining datasets with very large no-data areas, and thus it is optimal for combining the low-resolution bathymetric datasets with high-resolution survey line bathymetry. For this purpose, first a very high resolution bathymetric grid is constructed from the survey line bathymetry, which has valid bathymetric values given only along the survey lines and no-data areas everywhere else (i.e. in-between the survey lines and outside the surveyed region), and then this grid is used as one of the input datasets for the dataset composition. Furthermore, this technique fulfills the requirement to preserve the bathymetry values of the high-resolution dataset exactly (without being influenced by the compositing) while still achieving a smooth transition to the low-resolution dataset.

5.1. Seamless composition of incomplete bathymetric grids

The technique for seamless composition of multiple, potentially incomplete datasets presented in this section is based on the multi-band technique for the composition of image mosaics proposed by [BA83]. The basic idea behind multi-band blending of two images is to decompose each image into frequency bands. Each frequency band is combined separately using a weighting function that fits the size of the features in the respective band. The composite bands are finally

recombined to obtain the blended image. This technique allows overlapping images to be blended smoothly without introducing visible seams between the images while still preserving the high frequency details. Furthermore, in contrast to other blending techniques, it avoids ghosting or other artifacts that would make the transition recognizable in the composed dataset.

This technique can be transferred directly to the blending of height maps. However, the original algorithm assumes that the input maps overlap completely (since for the lowest frequency band, the width of the transition region in principle spans across the entire map), which means that it requires the input datasets to contain valid data everywhere. If this is not the case (such as for the incomplete bathymetric datasets that we are considering), the authors suggest to inter- or extrapolate the input maps (or their frequency bands) before the blending. However, this means that inter- or extrapolated data will be incorporated into the resulting composite map (within the transition regions of the respective frequency bands), which may be undesirable, at least in those cases, where the contents of a no-data area cannot be reasonably estimated based on its vicinity.

Therefore, we use another approach, which follows the basic concept that at no-data areas, the composite height should be calculated based only on those datasets that have valid data at the respective location, even inside the transition regions. This is realized by using a representation for the input maps that is similar to those of premultiplied alpha textures: For each input height map, we encode the location of no-data areas in an alpha map that contains value 0 inside the no-data areas and non-zero values elsewhere. Then, we premultiply the height maps with the corresponding alpha maps, which implies that inside no-data areas the height value 0 is stored. Then we perform the multi-band blending for the premultiplied height maps and for the alpha maps independently (for equivalent, arbitrarily shaped map regions). Finally, we divide the composite premultiplied height map entry-wise by the composite alpha map to obtain the final composite height map.

To guarantee smooth transitions despite the fact that blending within the no-data areas is avoided, high-frequency changes in the alpha maps and thereby induced high-frequency changes in the premultiplied height maps must be avoided. We do this by writing values smaller than 1 into the alpha map in the vicinity of no-data areas (and close to the map boundary). This can be done by creating a distance map for each considered height map, where each pixel is assigned the distance to the closest pixel for which no data is available in the respective height map. We compute corresponding alpha map values from the distances by dividing them by some maximum distance and limiting the result to the range 0 to 1. The maximum distance should be defined sufficiently large, but with the restriction that areas of valid data should contain no structures smaller than those distance.

5.2. Seamless integration of high-resolution data into low-resolution bathymetric datasets

While the technique described in the previous section can assure smooth transitions in case of no-data areas for datasets with medium-sized or thin no-data areas, smoothness cannot be fully guaranteed if the dataset has only thin structures of actual data values, such as in case of a bathymetric grid created from survey line bathymetry.

Furthermore, if this technique would be used for integrating the survey line bathymetry into the digital terrain model, at least the low frequency components of the survey line bathymetry values would be blended with the respective values from the other datasets (which would imply a global height adjustment of the survey line bathymetry according to the low resolution data). This is very undesirable in our setting, since we would loose the coincidence between the digital terrain model and the structural information contained in the seismic profile images (which is of relevance when visualizing both simultaneously).

For these reasons, we describe an alternative technique for integrating the high-resolution survey line bathymetry into low-resolution bathymetric datasets in the following. It has the property that it exactly preserves the bathymetry values for the integrated survey line bathymetry, and still a smooth transition to the surrounding low-resolution data is constructed. This means that the transition must take place completely outside the area where the high-resolution bathymetric grid that was generated from the survey line bathymetry has valid data. It is quite obvious that this can only be achieved by means of inter- or extrapolation of high-resolution data within the transition region.

After having constructed the high-resolution bathymetric grid, the algorithm is as follows: In a first step, we put the high-resolution and low-resolution bathymetry data in a common map without any blending, in such a way that high-resolution data is written into the map wherever it is available (and we do not want to discard it) and low-resolution data everywhere else. Additionally, we create a weight map that contains value 1 at those places where the combined map contains high-resolution data and values between 0 and 1 elsewhere. More precisely, the weight values should gradually decrease from 1 to 0 outside of the high-resolution data areas. This can be realized by constructing a map of (squared) distances to the high-resolution data areas, similar as described before.

Afterwards, we apply the pull-push algorithm according to [GGSC96] to the combined height map (using the associated weight map). This is an efficient and robust inter- and extrapolation technique that interprets the given weights as confidence values in the following sense: Weight 1 means that the corresponding value in the height map has maximum confidence and should thus not be changed; weight 0 means that it is invalid and should be replaced by a value interpolated from valid data in the vicinity; and in case of weights
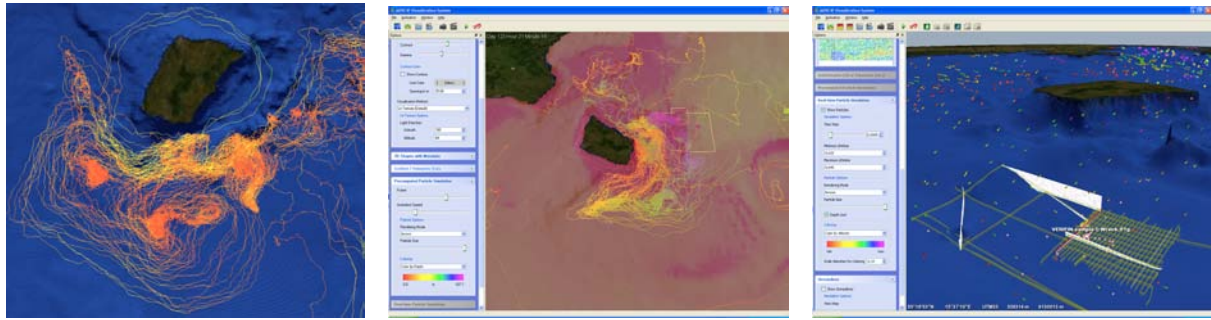
Figure 3: *Left: Particle trajectories. Center: Visualization of a PEC grid for a certain CWA release scenario and the currents assumed therein. Right: Visualizing a particle simulation, seismic profiles, network lines, and sample locations simultaneously.*

between 0 and 1, the new height value is to be constructed as a blend between the old value and an interpolated value.

This results in a height map with a smooth transition between high- and low-resolution bathymetry. The result can be further improved, if we smooth the weight map using morphological operators before performing the interpolation. This avoids abrupt changes of slope in the weight map as well as in the resulting blended dataset.

5.3. Results

Using the described techniques, we successfully generated a digital terrain model from a combination of the different datasets that has valid and plausible heights all over the domain and no visible seams or otherwise noticeable transition areas in it, see Figure 1. When visualizing the seismic profiles that have been measured along the same survey lines as the bathymetry in the form of 3D curtains on the final digital terrain model, into which the survey line bathymetry has been integrated, one can clearly see that the terrain model and the seismic measurements coincide exactly.

6. Conclusion

We presented an efficient innovative visualization framework with special focus on the appropriate integration of heterogeneous geospatial data of varying extents and varying resolution. Altogether, it provides a diversity of real-time visualization capabilities not found in prior geospatial visualization tools. In addition, existing techniques have been improved with respect to visual quality and run-time performance. By using generic techniques where possible, we tried to ensure a broad applicability of the developed system.

Future work. The developed application was already successfully employed by scientists of different disciplines in context of the MERCW project. A usability study with a wider range of users is currently under way. Furthermore, we plan to employ and extend the system also for visualization tasks in other interdisciplinary research projects.

References

[BA83] BURT P. J., ADELSON E. H.: A multiresolution spline with application to image mosaics. *ACM Trans. Graphics 2* (Oct. 1983), 217–236.

[GGSC96] GORTLER S. J., GRZESZCZUK R., SZELISKI R., CO-HEN M. F.: The Lumigraph. *Comp. Graphics 30* (1996), 43–54.

[KKKW05] KRÜGER J., KIPFER P., KONDRATIEVA P., WEST-ERMANN R.: A particle system for interactive visualization of 3D flows. *IEEE Trans. Visualization and Computer Graphics 11*, 6 (Nov. 2005), 744–756.

[MPG*00] MAYER L. A., PATON M., GEE L., GARDNER J. V., WARE C.: Interactive 3D visualization: A tool for seafloor navigation, exploration and engineering. In *Proc. Oceans 2000 MTS/IEEE Conference* (Oct. 2000), vol. 2, pp. 913–920.

[OSG] OSGEO FOUNDATION: GDAL – Geospatial Data Abstraction Library. http://www.gdal.org/.

[SK07] SCHNEIDER M., KLEIN R.: Efficient and accurate rendering of vector data on virtual landscapes. *Journal of WSCG 15*, 1-3 (Jan. 2007).

[SML06] SCHROEDER W., MARTIN K., LORENSEN B.: *The Visualization Toolkit: An Object-Oriented Approach to 3D Graphics*, 4th ed. Kitware, Inc., Dec. 2006.

[TBF00] THOMPSON D., BRAUN J., FORD R.: *OpenDX: Paths to Visualization*. VIS, Inc., 2000.

[UCA] UCAR/UNIDATA: Common Data Model. http://www.unidata.ucar.edu/software/netcdf-java/CDM/.

[WK07] WAHL R., KLEIN R.: Towards semantic interaction in high-detail realtime terrain and city visualization. In *PIA07: Photogrammetric Image Analysis* (Sept. 2007), vol. 36, pp. 179–184.

[WMD*04] WAHL R., MASSING M., DEGENER P., GUTHE M., KLEIN R.: Scalable compression and rendering of textured terrain data. *Journal of WSCG 12*, 3 (Feb. 2004), 521–528.

[WPA*01] WARE C., PLUMLEE M., ARSENAULT R., MAYER L. A., SMITH S., HOUSE D.: GeoZui3D: Data fusion for interpreting oceanographic data. In *Proc. Oceans 2001 MTS/IEEE Conference* (Jan. 2001), vol. 3, pp. 1960–1964.