Ovis: A Framework for Visual Analysis of Ocean Forecast Ensembles

Thomas Höllt, *Member, IEEE,* Ahmed Magdy, Peng Zhan, Guoning Chen, *Member, IEEE,* Ganesh Gopalakrishnan, Ibrahim Hoteit, Charles D. Hansen, *Fellow, IEEE,* Markus Hadwiger, *Member, IEEE*

Abstract—We present a novel integrated visualization system that enables interactive visual analysis of ensemble simulations of the sea surface height that is used in ocean forecasting. The position of eddies can be derived directly from the sea surface height and our visualization approach enables their interactive exploration and analysis.

The behavior of eddies is important in different application settings of which we present two in this paper. First, we show an application for interactive planning of placement as well as operation of off-shore structures using real-world ensemble simulation data of the Gulf of Mexico. Off-shore structures, such as those used for oil exploration, are vulnerable to hazards caused by eddies, and the oil and gas industry relies on ocean forecasts for efficient operations. We enable analysis of the spatial domain, as well as the temporal evolution, for planning the placement and operation of structures.

Eddies are also important for marine life. They transport water over large distances and with it also heat and other physical properties as well as biological organisms. In the second application we present the usefulness of our tool, which could be used for planning the paths of autonomous underwater vehicles, so called gliders, for marine scientists to study simulation data of the largely unexplored Red Sea.

Index Terms—Ensemble Visualization, Ocean Visualization, Ocean Forecast, Risk Estimation.

1 Introduction

OCEAN forecasts are widely used for decision making in a large range of areas. The oil and gas industry relies on forecasts to safely operate off-shore structures for oil exploration. Strong currents, such as eddies, could severely affect the operations of these platforms. Marine scientists acquire data using autonomous underwater gliders whose paths can be optimized when currents are known beforehand.

Nowadays, these forecasts do not come as single simulation results, but as ensembles of simulations, mapping uncertainty in the starting conditions, as well as the simulation models themselves, to variation in the ensemble. Developing efficient tools to visualize and clearly disseminate such forecast outputs and results is becoming a very important part of the forecasting process. Such tools have to be conceived in a way that allows users to easily extract and clearly identify the necessary information from large ensembles and the associated statistics representing the forecast and its uncertainties.

In earlier work [14] we present the first integrated system for the visual exploration and analysis of these kinds of forecasts. Our system handles time-series of multivalued ensembles of sea surface height data, represented by 2D heightfields. A set of statistical properties is derived from the ensemble and can be explored in multiple linked views, while the complete ensemble is always available for detailed inspection on demand. Our system enables domain experts to efficiently analyze ocean forecasts, including their corresponding uncertainties. In this work we extend our framework by a more advanced technique for eddy classification, which is used in a second application scenario, also presented in this paper. We present the application of our framework in two settings. First for planning the placement and operations of off-shore structures, such as oil platforms in the Gulf of Mexico, and second, new to this work, to aid planning the paths of underwater gliders in the Red Sea.

1.1 Ocean Forecast Simulation

The development of a reliable ocean forecasting system requires models capable of simulating ocean circulation and an efficient assimilation scheme that, given enough observations, provides accurate initial conditions for forecasting. High-resolution 3D general circulation ocean models are necessary to reproduce complex mesoscale dynamics like in the Gulf of Mexico [3]. However, such models cannot provide accurate forecasts of mesoscale variability, such as eddy

- *T. Höllt and M. Hadwiger are with the Geometric Modeling and Scientific Visualization Center, King Abdullah University of Science and Technology, Thuwal, Saudi Arabia.*
  E-mail: {thomas.hollt|markus.hadwiger}@kaust.edu.sa
- *A. Magdy is with Wireless Stars, Cairo, Egypt.*
  E-mail: ahmedmagdy.mohamed@kaust.edu.sa
- *P. Zhan and I. Hoteit are with the Earth Science and Engineering Department, King Abdullah University of Science and Technology, Thuwal, Saudi Arabia.*
  E-mail: {peng.zhan|ibrahim.hoteit}@kaust.edu.sa
- *G. Chen is with the Department of Computer Science, University of Houston, TX.*
  E-mail: chengu@cs.uh.edu
- *G. Gopalakrishnan is with the Scripps Institution of Oceanography, University of California, San Diego, CA.*
  E-mail: ggopalak@ucsd.edu
- *C.D. Hansen is with the SCI Institute and School of Computing at the University of Utah, Salt Lake City, UT.*
  E-mail: hansen@cs.utah.edu

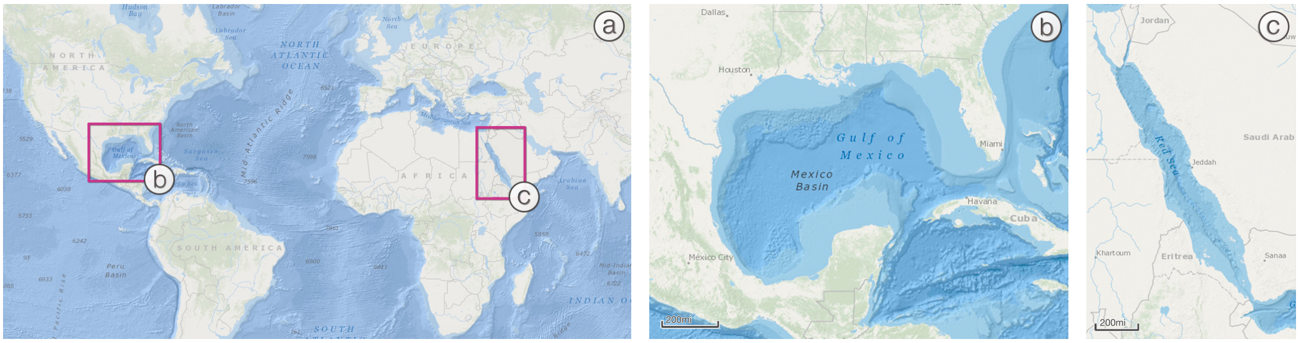


Fig. 1. **Maps of the Areas of Interest.** The Gulf of Mexico area is shown in (b). The simulation area for the Red Sea dataset extends beyond the area illustrated in (c). However, for the use case that we present in Section 5.2, only the area shown in (c) is of interest. Map data courtesy of GEBCO, IHO-IOC GEBCO, NGS, DeLorme.

shedding events, without data assimilation. A general circulation ocean model is subject to several sources of uncertainties, not only from the poorly known inputs such as the initial state, and atmospheric and lateral boundary conditions, but also from the use of approximate parameterization schemes of sub-grid physics and ocean mixing dynamics. Data assimilation methods address this issue by constraining model outputs with incoming data.

The important role of uncertainties is now increasingly recognized in the ocean prediction community for proper decision making and risk management.

New assimilation methods based on Bayesian filtering theory have been recently developed by the ocean and atmospheric communities for efficient propagation and quantification of uncertainties [7], [15], [16], [17], [18], [33]. These methods, known as ensemble Kalman filter methods, follow a Monte Carlo approach to represent the uncertainties on a state estimate by an ensemble of model states. These are then integrated forward in time with the general circulation ocean model to quantify uncertainties in the forecast. The estimated forecast uncertainties are then combined with the observation uncertainties to assimilate the new incoming data using a Kalman filter correction step [7], before a new forecast cycle begins. Developing and implementing efficient ensemble Kalman filters with state-of-the-art ocean and atmospheric models is a very active area of research.

With the fast-growing high performance computing resources, the implementation of ensemble Kalman filters with large ensemble members is now practically feasible using highly sophisticated general circulation ocean models. When a filter's ensemble is available, it is customary to calculate various statistical measures of the ensemble spread as indicators of the uncertainties and of their evolution in space and time, which are then used in decision making.

Hoteit et al. [15] developed an ensemble forecasting system for the Gulf of Mexico circulation based on the Massachusetts Institute of Technology General Circulation Model (MITgcm) [25], and the Data Assimilation Research Testbed (DART) [18]. This system is capable of assimilating various sets of satellite and in-situ ocean observations. A similar system covering the Red Sea was developed very recently. We use these systems as real-world scenarios that illustrate the new capabilities for analysis and exploration provided by our visualization approach. Figure 1 gives an overview of the areas covered by the forecasting systems.

1.2 Visualization Contributions

We present a GPU-based interactive visualization system for the exploration and analysis of ensemble heightfield data, with a focus on the specific requirements of ocean forecasts. Based on an efficient GPU pipeline, we perform on-the-fly statistical analysis of the ensemble data, allowing interactive parameter space exploration. Usually these kinds of data are visualized by means of parameterization, e.g. fitting a Gaussian curve and storing only σ and μ. This requires a priori knowledge of the data, i.e. the distribution of ensemble members must correspond to a normal or at least unimodal distribution. One key difference of our approach is that we do not assume any such properties. The whole dataset, or at least the distribution by means of a histogram, is available throughout the pipeline. This allows us to carry out visualization and computation such as iso-contour extraction on the original data.

Based on our framework we present a novel workflow for planning the placement and operation of off-shore structures such as oil rigs as well as for planning the paths of autonomous sea vehicles, such as gliders used for data acquisition in marine research. While we focus on the visualization and analysis of ocean forecast data, the presented framework could also be used for the exploration of heightfield ensembles from other fluid earth systems, such as weather forecasting or climate simulation, but also completely unrelated fields, such as for exploration of ensemble segmentation data [13].

2 Related Work

Uncertainty and ensemble visualization are widely recognized as important topics in the field of visualization, which has resulted in a large body of related work in recent years. In the following overview, we restrict ourselves to key publications in uncertainty and ensemble visualization, as well as selected publications from other areas related to the techniques presented in this paper.

Uncertainty Visualization. A good introduction to uncertainty visualization is provided by Pang et al. [30], who present a detailed classification of uncertainty, as well as numerous visualization techniques. Johnson and Sanderson [19] give a good overview of uncertainty visualization techniques for 2D and 3D scientific visualization, including uncertainty in surfaces. For a definition of the basic concepts of uncertainty and another overview of visualization techniques we refer to Griethe and Schumann [9]. Riveiro [40] provides an evaluation of different uncertainty visualization techniques for information fusion. Rhodes et al. [39] present the use of color and texture to visualize uncertainty of iso-surfaces. Brown [1] employs animation for the same task. Grigoryan and Rheingans [10] present a combination of surface and point-based rendering to visualize uncertainty in tumor growth. Uncertainty information is provided by rendering point clouds in areas of large uncertainty, as opposed to crisp surfaces in more certain areas.

In recent work Pöthkow et al. [35], [36] as well as Pfaffelmoser et al. [31] present techniques to extract and visualize uncertainty in probabilistic iso-surfaces. Pfaffelmoser and Westermann [32] describe a technique for the visualization of correlation structures in uncertain 2D scalar fields. They use spatial clustering based on the degree of dependency of a random variable and its neighborhood.

Saad et al. [41] present a system which models and visualizes uncertainty in segmentation data based on a priori shape and appearance knowledge.

Ensemble Visualization. Pang, Kao and colleagues present early work on visualization of ensemble data [20], [21], [23], [24]. While the authors do not use the term ensemble, these works deal with the visualization of what they call spatial distribution data, which they define as a collection of n values for a single variable in m dimensions. These are essentially ensemble data. The authors adapt standard visualization techniques to visualize these data gathered from various sensors, e.g. satellite imaging or multi-return Lidar. Frameworks for visualization of ensemble data gained from weather simulations include *Ensemble-Vis* by Potter et al. [38] and *Noodles* by Sanyal et al. [44]. These papers describe fully featured applications focused on the specific needs for analyzing weather simulation data. They implement multiple linked views to visualize a complete set of multidimensional, multivariate and multivalued ensembles. While these frameworks provide tools for visualizing complete simulation ensembles including multiple dimensions, to solve the problem presented in this work we focus on heightfield ensemble data.

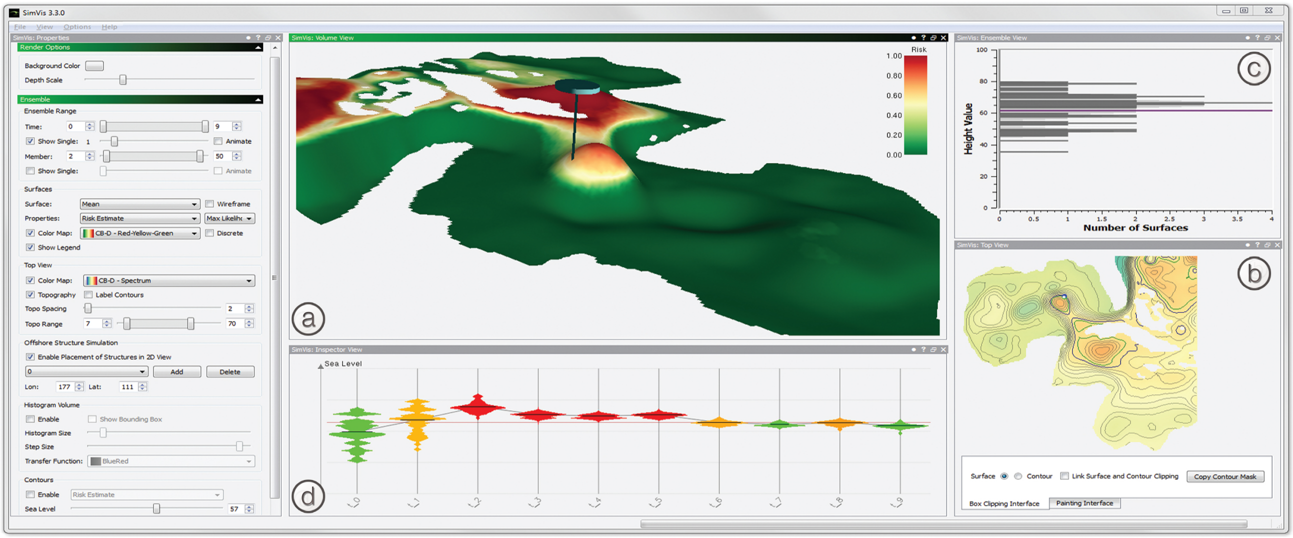
Matković et al. [26] present a framework for visual analysis of families of surfaces by projecting the surface data into lower-dimensional spaces. Piringer et al. [34] describe a system for comparative analysis of 2D function ensembles used in the development process of powertrain systems. Their design focuses on comparison of 2D functions at multiple levels of detail. Healey and Snoeyink [11] present a similar approach for visualizing error in terrain representation. There, the error, which can be introduced by sensors, data processing or data representation, is modeled as the difference between the active model and a given ground truth.

Several published extensions of box plots have inspired our time-series view. Hintze and Nelson [12] introduce violin plots to give an indication of the distribution using the sides of the box. Esty and Banfield [6] combine box and percentile plots to add the complete distribution to the plot while keeping the simplicity of box plots. Potter et al. [37] combine quartile, moment and density plots, based on the histogram, to create summary plots. The density of curves in 1D function plots can be visualized effectively using kernel density estimation [22]. Our histogram view that shows the distribution of surfaces embedded in 3D passing through each (x, y) position is similar in spirit to such approaches, but for primitives of one dimension higher.

3 System Design

This section describes our system for the visual analysis of multivalued sea surface height data gathered from ocean forecasts. The foundation for the visual exploration of the data comprises an extensive analysis of the data, which is described in Section 3.1. We compute a set of standard statistics to indicate variation at each grid point over the ensemble. For both of the applications that we present in Section 5, position and movement of eddies is of particular interest. The position of these eddies can directly be derived from the sea surface height. In addition to the statical analysis we compute the eddy centers for all ensemble members as well as the probability of belonging to an eddy for each grid point.

For efficient exploration of the data we provide multiple linked views (Section 3.2) as shown in Figure 2. Two spatial views, in 3D (Figure 2(a)) and 2D (Figure 2(b)) provide an overview of the spatial relationships. The 2D view is also used for defining the area of interest as well as defining potential platform positions and glider paths. For a more detailed inspection we provide a histogram view (Figure 2(c)) for a single position in time and space as well as a time-series view (Figure 2(d)) which provides detailed


SimVis 3.3.0

SimVis: Properties

Render Options

Background Color

Depth Scale

Ensemble

Ensemble Range

Time: 0 – 9

☑ Show Single: 1 □ Animate

Member: 2 – 50

□ Show Single: □ Animate

Surfaces

Surface: Mean □ Wireframe

Properties: Risk Estimate Max Likelih

☑ Color Map: CB-D - Red-Yellow-Green □ Discrete

☑ Show Legend

Top View

☑ Color Map: CB-D - Spectrum

☑ Topography □ Label Contours

Topo Spacing 2

Topo Range 7 – 70

Offshore Structure Simulation

☑ Enable Placement of Structures in 2D View

0 Add Delete

Lon: 177 Lat: 111

Histogram Volume

□ Enable □ Show Bounding Box

Histogram Size

Step Size

Transfer Function: BlueRed

Contours

□ Enable Risk Estimate

Sea Level 57

Surface ● ○ Contour □ Link Surface and Contour Clipping Copy Contour Mask

Box Clipping Interface Painting Interface

Fig. 2. **System Overview.** Our system for exploration of ocean forecast ensembles consists of four main views. The simulated ocean surface, or a derived version like the mean surface for a single point in time, can be shown in 3D or 2D (a) and (b). The histogram view (c) shows the complete distribution of the ensemble at a selected position, while the time-series view (d) shows the distribution and the resulting operational risk at a selected position for multiple samples along time.

information for a user-defined set of positions in time and space.

The analysis of the data is completely visualization-driven. Statistical properties are computed on the fly only when needed and for any user-defined subset of the ensemble, to allow inspection of the parameter space of the simulation. Using a completely GPU-based computation and visualization pipeline as presented in Section 4 allows updates of these data at interactive rates.

3.1 Analysis

Statistical Analysis. The goal of the statistical analysis is to provide information on the distribution of the members within the ensemble. Therefore, for each grid point, we compute a histogram of the sea surface height at this position over all ensemble members. Based on the histograms we compute a kernel density estimation to approximate the continuous probability density function (pdf). For visualization we assemble the 1D pdfs in a 3D volume by placing each pdf at the appropriate position in the grid.

For each grid point a number of statistical properties including range, mean, median, maximum mode, standard deviation, variance, skewness and kurtosis are computed, based on the 1D histograms. Similar to the probability density functions these values are then assembled in the original grid to form 2D scalar fields, which are divided into two groups of different semantics. While mean, median and maximum mode are treated as additional, synthetic surfaces, the other properties are added as meta information, for example to color-code the surfaces.

Eddy Tracking. A very simple approach to automatically identifying eddies based on the sea surface height only is to look at the absolute height values. Clockwise rotating eddies will push the water up while counterclockwise rotation will lead to a drop in sea surface height. Based on a threshold, each grid point can be classified as belonging to an eddy or not. In our application this threshold is user-defined and can be modified at any time. The classification as well as the resulting risk estimate (see below) for the complete ensemble will be updated accordingly on the fly.

Chaigneau et al. [2] present a more advanced technique for automatic eddy identification. Their method is based on actual flows on the sea surface. Chaigneau et al. use satellite measurements of the sea surface height as input for their method. In absence of a measured velocity field they derive the geostrophic velocity field [8] $(U(k,x,y,t), V(k,x,y,t))$ from the gradients of the sea surface height $H(k,x,y,t)$ as:

$$U = -\frac{g}{f} \cdot \frac{\partial H}{\partial y} \tag{1}$$

$$V = \frac{g}{f} \cdot \frac{\partial H}{\partial x}, \tag{2}$$

with g the gravitational acceleration and f the Coriolis parameter, k the ensemble member, x and y the spatial dimensions and t the temporal dimension. While Chaigneau et al. are constrained by the fact that they only use the measured sea surface height as input, any velocity field representing the surface flow, such as results from the simulation, can be used as input. When using the geostrophic velocity field, the extracted streamlines match to iso-contours of H.

With given sea surface height and velocity fields the method consists of four steps:

1) Using a moving window approach, local extrema in H are tagged as potential eddy centers.
2) Streamlines on the velocity field, around each potential eddy center, are computed. Closed streamlines, i.e. taking a full 360° turn, within a limited area and with limited wiggling, indicate the characteristic circular eddy currents. For this we employ the winding angle algorithm as presented by Sadarjoen et al. [42], [43].
3) Extrema for which no closed streamline can be found, such as extrema close to the boundary of the dataset, are dropped from the list of potential eddy centers. The remaining candidates will be considered eddy centers.
4) Closed streamlines for each eddy center are clustered. The outermost streamline for each eddy center identifies the boundary of the eddy.

Flood-filling the outermost streamlines classifies each grid point inside as corresponding to an eddy.

Risk Estimate. Both eddy tracking methods introduced above result in a binary map $M(k,x,y,t)$ for each member k of the ensemble that is true if $M(k,x,y,t)$ is classified as an eddy and false otherwise. Assuming that all realizations for one sample in time (assimilation cycle) t are equally probable, we define the eddy probability $P_e(x,y,t)$ or risk for each position in space and time as the fraction of the members in the ensemble where the bit is set in these maps for this position:

$$P_e(x,y,t) = \frac{1}{n} \cdot \sum_{k=1}^{n} \begin{cases} 1 & \text{if } M(k,x,y,t) \text{ is true} \\ 0 & \text{else,} \end{cases} \tag{3}$$

where n is the total number of realizations for a single assimilation cycle.

This is especially important for the oil rig-operation application we present in Section 5.1. The strong currents of eddies pose large risks to deep sea oil exploration. Hence we also call this property *risk estimate*. In addition to the uncertainty information that can be gathered for each position, the risk estimate map also provides an idea of the positional uncertainty. This can clearly be seen in Figure 3. Eddies that are at similar positions and of similar size over all members at a single point in time have sharp boundaries (a), while eddies with a lot of variation will show up in the visualization with fuzzy boundaries (b).

3.2 Visual Forecast Exploration

Our system targets the interpretation of forecasts for different applications. Since the different applications have different requirements, we provide a set of four main views, which are used in different combinations depending on the application scenario. Figure 2 shows our application with the main views plus a unified settings panel. The views are two spatial views showing the surface data themselves, one in 3D (a), the other one in 2D (b), a linked histogram view (c) as well as a time-series view (d).

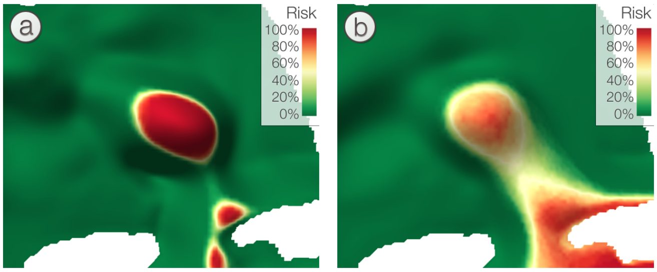


Fig. 3. **Spatial Uncertainty** can be seen using the risk estimate. The two figures show the risk for the same region but different assimilation cycles. In (a) the eddy can be classified with little uncertainty, indicated by the sharp features. In (b) the fuzzy boundary indicates larger spatial uncertainty.

2D View. The simple 2D view shown in Figure 2(b) is a common tool for visualizing heightfield data and familiar to domain scientists.

The main function of the view is to provide a first overview of the data. Therefore it provides two ways to visualize 2D scalar fields. The scalar field can be visualized directly by pseudo coloring the scalar values, or indirectly by extracting iso-contours from the scalar field which can then be rendered as curves in this view. Typically the topography of the mean sea surface height for a single point in time is rendered by extracting iso-contours for several interesting height values. Uncertainty information can be provided for example by rendering the variance scalar field in the background. In general the view is completely user configurable. Any of the scalar fields, including any of the original heightfields from the ensemble, can be rendered directly or by means of iso-contouring.

In addition to presenting information to the user, the 2D view is also used for interaction. We provide a simple painting interface in this view to allow the definition of an area of interest. The user can select rectangles or directly paint interesting regions on the map, allowing arbitrary free-form selections, which will then be highlighted in both the 2D and 3D views. Furthermore the view enables the creation and editing of points and paths of interest. This allows probing positions, which could be potential positions for placing an off-shore structure (see Section 5.1) or defining and adjusting the path for a glider (Section 5.2).

3D View. The 3D view (Figure 2(a)) provides the same features for visualization as the 2D view plus several additional tools for a more detailed spatial and temporal inspection. Typically, in the 3D view the height values of the displayed surface are mapped to the third dimension, freeing pseudo-coloring and iso-contours for additional information. An additional benefit of the 3D view is that it is possible to use volume rendering for showing details of the distribution of the ensemble. Similar to approaches presented by Pöthkow et al. [35], [36], as well as Pfaffelmoser

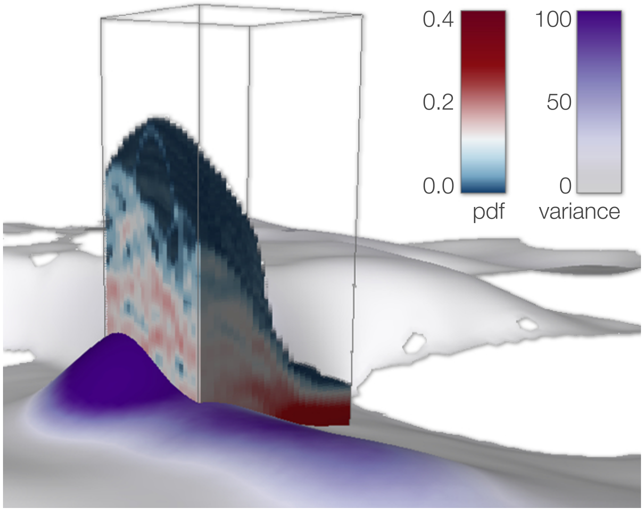


Fig. 4. **3D Probability Density Function** at a user-selected position. The surface is color-mapped with the variance. The large spread in areas of high variance is clearly visible in the volume rendering.

et al. [31], we depict the actual distribution of the ensemble as a volume around the surface. Instead of using a parametric representation of the data based on mean and variance, we allow rendering the full probability density function, of the distribution, as presented in Section 3.1, to allow detailed inspection of the actual data. However, since at this point the user usually has picked a set of points of interest, to avoid unnecessary occlusion, we do not render the complete volume, but a small subset of adjustable size, which essentially works like a volumetric cursor (see Figure 4). The user can simply probe the data by hovering with the mouse over a position of interest, and the probability density volume is then rendered around the picked position.

Histogram View. Another way to inspect the distribution in detail is the histogram view shown in Figure 2(c). This view shows the histogram of all height values of the ensemble as well as the probability density function for a selected (x, y)-position. To provide spatial context the histogram is positioned according to the actual height values and the bin corresponding to the surface that is currently shown in the 3D view is highlighted. The histogram is laid out with the sea surface height on the y-axis and the number of surfaces on the x-axis. This layout deviates from the conventional layout for a histogram where usually the bins are mapped to the horizontal axis, but in our case the sea surface height intuitively maps to the vertical axis. Similar to the volumetric representation in the 3D view, the position is defined by picking directly in any of the spatial views. When the user moves the mouse over the surface, the histogram view is updated on the fly to show the histogram at the current mouse position.

Time-Series View. The time-series view (Figure 2(d)) provides detailed information on single positions over time, by means of a glyph, specifically designed to convey the most important information for the applications presented in Section 5. The glyphs are positioned side by side, along the x-axis of the view. In general, each of these glyphs can show data from any desired point in time and space. We will show in section 5 how this can be used to provide different semantics for this view depending on the application. For planning the operations of an oil platform at a fixed position the x,y coordinate is fixed, while each glyph presents the information for a sample along the timeline. For planning the glider paths the path is sampled along the time-series, providing distinctive x,y-coordinates for each point in time.

Figure 5 shows a detailed description of the glyph, which is inspired by the violin plots, introduced by Hintze and Nelson [12]. While the shape of the violin plot is symmetric, the left and the right side of our glyph can be defined by two different properties. In the example in Figure 5, the left side shows the pdf for the distance to the closest eddy center from the position of the glyph, while the right side shows the pdf of the sea surface height as described in Section 3.1. For the different applications the glyph can be configured by the user, for example for comparing two positions for placement the pdfs of the two positions can be used on the two sides of the glyph. However, the most important application is displaying sea surface height distributions. Therefore we decided to use the same vertical layout as described above for the histogram view. The glyph is positioned on the y-axis according to the actual height values, making the position not only comparable to other glyphs at different positions in the view, but also to the user-defined threshold for

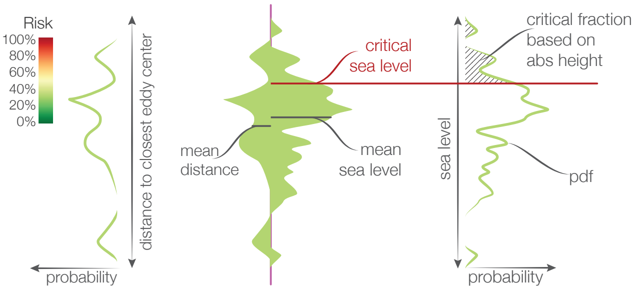


Fig. 5. The **Time-Series View Glyph** in detail. Here the left side of the glyph shows the pdf of the distance to the closest eddy center for all members of the ensemble, while the right side shows the pdf of the sea surface height. The mean values are indicated by a bar at the appropriate position. Colorcoding is used to color the glyph according to the risk at the selected position. The sides of the glyph can be configured by the user, for example to show the sea surface height at two different positions, one on each side of the glyph.

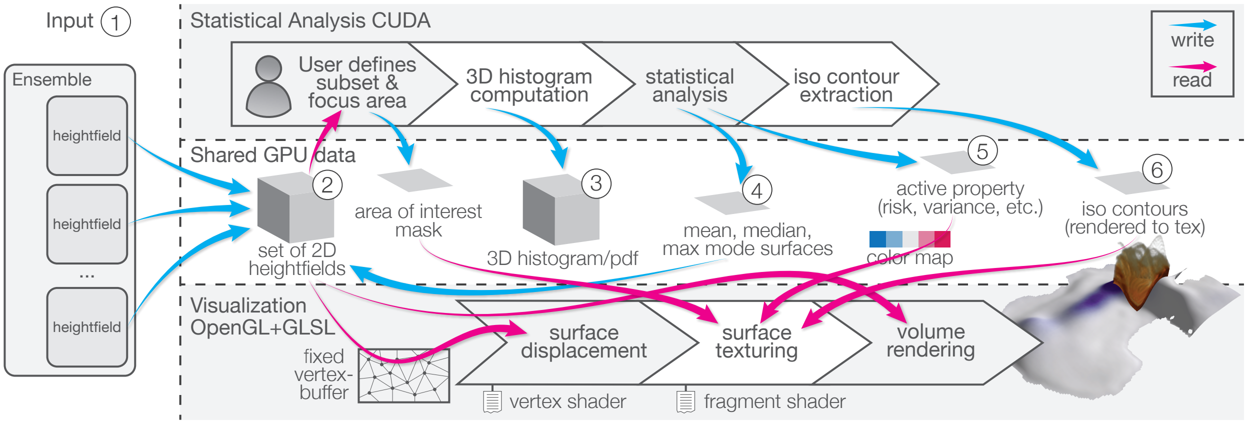


Fig. 6. **Pipeline Overview.** The pipeline is divided into two major blocks: The statistical analysis part at the top, and the rendering part shown at the bottom. Both parts are entirely GPU-based, and all data (middle row) are shared by both parts in GPU memory.

the simple, absolute height-based eddy classification, indicated by the horizontal red line. The mean values of both properties are indicated by a black bar on their respective sides of the glyph. Additionally, the glyphs are pseudo-colored according to one of the two versions of the risk estimates described in Section 3.1. Information on the uncertainty of the data can immediately be retrieved from the shape of the glyph: A large spread or variation in the surface positions over the ensemble, indicating larger uncertainty, results in a large glyph, while little uncertainty results in less variation and more compact glyphs.

4 GPU-Based Pipeline

To allow interactive updates of the statistics, we have implemented a completely GPU-based analysis and visualization pipeline presented in Section 4.1. A detailed performance analysis is presented in Section 4.2.

4.1 Implementation

Our GPU-based analysis and visualization pipeline is illustrated in Figure 6. In the remainder of this section, numbers in brackets refer to this figure. The pipeline is divided into two main parts: The statistical analysis and iso-surface extraction is carried out using CUDA, while the visualization is based on OpenGL and GLSL shaders. All data are shared between the two parts of the pipeline, so that after the initial upload of the ensemble onto the GPU no expensive bus transfer is necessary. Since usually only a small part of the ensemble is required by the visualization, a streaming approach would be possible for datasets that are larger than GPU memory, but we currently assume that the dataset fits into GPU memory.

Input. The input (1) to our system is a set of heightfields. These can be part of a simulation ensemble, e.g. from ocean or weather forecasts, a time-series of some sort, or the results of a parameterized segmentation [13]. Even though we focus on heightfields in this work, the concepts can also be applied to surfaces in n dimensions as long as the correspondences between all surfaces in the dataset are known for every nD-datapoint. In our framework, we assume the 2D spatial (x, y)-coordinate to be the correspondence between the surfaces.

Data Representation. Before computation of statistics or visualization, the ensemble is converted into a 3D texture (2) and loaded onto the GPU. Every heightfield of the ensemble will be represented by one slice in this texture. Additionally, space for the mean, median and maximum mode heightfield will also be reserved in this texture. The surfaces are indexed using the original parametrization. If there is only a single parameter, for example the temporal samples in a time series, the surface ID corresponds to the texture index. For higher-dimensional parameter spaces, e.g. ensemble ID plus time, the linear texture index is computed from the original parameters. This allows the user to define subranges for each parameter separately, for example to examine the complete ensemble at a single point in time.

Statistical Analysis. The first step in the statistical analysis is the creation of the 3D histogram (3). Changes in the parameter range trigger an update of the 3D histogram and subsequently of the representative surface and property texture. Since each ensemble member provides exactly one entry to the histogram per (x, y)-position, rather than using a thread for each member, we use one thread per (x, y)-position. Each thread then loops over all selected surfaces and inserts the corresponding height values into the histogram. This way, write conflicts can be avoided and no critical sections or atomic operations are needed. The kernels for the derived properties are set up in a similar fashion. The desired statistical property is computed by one thread per (x, y)-position. The main difference to the histogram computation is that this results in a single scalar per thread, all of which are then assembled into a 2D texture. While mean, median and maximum mode (4) are attached to the 3D heightfield texture to be used as representative surfaces, the other properties (5) are copied into a 2D texture available to the visualization pipeline for texturing the surface. Exploiting the parallelism of the GPU and eliminating costly bus transfers between CPU and GPU allows interactive modification of the parameter range even for ensembles containing several hundred surfaces. Section 4.2 provides a detailed performance analysis.

Iso Contouring. We have implemented marching squares using CUDA, based on the marching cubes example from the CUDA SDK [28]. We keep the initial geometric representation of the contours, for example for use in the 2D view, but for overlaying the contours onto the 3D surfaces we render the contours into an offscreen buffer (6), which is then used for texturing.

Rendering. The rendering pipeline takes advantage of the fact that all ensemble data are already stored in GPU memory, which facilitates efficient surface rendering. Instead of creating new surface geometry every time a different surface of the ensemble is rendered, a single generic vertex buffer of fixed size is created. This buffer covers the entire (x, y)-domain, but does not contain any height information. The z-value of each vertex is set later in the vertex shader. Before transforming the vertex coordinates into view space, the object space (x, y)-coordinates of the vertex in combination with the ID of the active surface are used to look up the z-value of the current vertex in the ensemble texture. At this point, the desired surface geometry is available. In order to be able to visualize the results of the statistical analysis, the object space coordinates are attached to each vertex as texture coordinates (x and y are sufficient). In the fragment shader, this information can then be used to look up the active statistical property in the 2D texture. This texture contains the raw information from the statistical analysis, which is then converted to the fragment color by a look up in a 1D color map. We provide a selection of several continuous, diverging cool-to-warm color maps, as presented by Moreland [27], but also allow the creation of custom color maps. These color maps minimally interfere with shading, which is very important in this case, as shading is an important feature to judge the shape of a surface. During testing we realized that using the continuous version made it very hard to relate an actual value to a color in the rendering so we decided to optionally provide a discrete version with ten steps. After the surface geometry has been rendered, a surrounding volume, for example the 3D probability density function, can be rendered as well. This is done in a second rendering pass in order to guarantee correct visibility [45].

Interaction. With the described pipeline in place, a number of features can be implemented very easily and efficiently. If desired, the user can choose to render any surface from the ensemble. This requires no data transfer to or from the GPU, except for the ID of the surface in the ensemble to render. In addition, it is possible to automatically animate all surfaces in a predefined range. In the presented application this can be useful in two ways; As shown by Brown [1] animation is a powerful tool for visualizing uncertainty. The user can choose to animate through all members of a single sample of the time series to get an impression of the surface distribution. Secondly animating the mean surfaces over the time domain can show the behavior of the eddies.

The described visualization techniques can give a very good impression of the quantitative variation in the data. Detailed information on the surface distribution can be gained by animating through or manually selecting individual surfaces from the ensemble. However, it is hard to get a good impression of the complete distribution this way. We therefore provide the possibility to render a cutout of adjustable size of the pdf of the complete distribution as a volume on top of the surface geometry (Figure 4) or show histogram and pdf for a selected position in a separate view (Figure 2(c)). The position to investigate can be picked directly in the 3D view. All information that is required for picking is already available in our rendering pipeline: We use the same vertex shader as described before for rendering the surface into an off-screen buffer of the same size as the frame buffer. Instead of using the object space coordinates to look up the scalar values in the fragment shader, we use the coordinates directly as the vertex color. This way, we can look up the current mouse position directly in the downloaded off-screen buffer. With the (x, y)-components of the resulting volume position, we can then directly look up the histogram and probability density distribution for this position. To facilitate easy comparison, we color the bin corresponding to the current representative surface differently than the remaining bins.

4.2 Performance

The performance of the statistical analysis is crucial for interactive exploration of the parameter space. We have used the dataset of the Gulf of Mexico, described in Section 5 for a performance analysis. The dataset consists of a total of 500 surfaces spread over ten sampled time steps. Since usually one time step is investigated at a time, we compare performance for a single forecast time step, consisting of 50 surfaces, as well as the complete dataset. Table 1 shows the resulting computation times.

The computations were performed using an NVIDIA GeForce GTX 580 with 1.5GB of graphics memory. The timings were averaged over 1000 kernel executions. As all data stays on the GPU, no bus transfer has to be considered. For comparison, we also show computation times of a single time step on the CPU. The computations were carried out on a workstation with two six-core Xeons (12 physical

TABLE 1
Computation times for all properties. All times are given in milliseconds. The first column shows ID and name of the property. The columns titled *w/o* and *w dep* show the computation time for a property with and without dependencies. The last two columns show the speedup from CPU to GPU.

		50 Surfaces GPU		500 Surfaces GPU		50 Surfaces CPU		Speedup	
Property		w/o	w dep	w/o	w dep	w/o	w dep	w/o	w dep
1	Histogram	3.23	3.23	38.56	38.56	19.24	19.24	6.0x	6.0x
2	PDF^{1}	12.93	16.16	12.78	51.34	45.70	64.94	3.5x	4.0x
3	Range	0.71	0.71	11.09	11.09	3.45	3.45	4.9x	4.9x
4	Mean	0.71	0.71	10.89	10.89	3.48	3.48	4.9x	4.9x
5	Median^{1}	0.70	3.93	0.70	39.26	8.78	28.02	12.5x	7.1x
6	Mode^{1}	1.40	4.63	1.41	39.97	4.65	23.89	3.3x	5.2x
7	Variance^{4}	0.72	1.43	10.87	21.76	3.85	7.33	5.3x	5.1x
8	Std Dev^{4,7}	0.02	1.45	0.02	32.78	0.14	7.47	7.0x	5.2x
9	Skewness^{1,4,6,7,8}	0.05	6.13	0.05	72.80	0.16	31.42	3.2x	5.1x
10	Kurtosis^{4,7}	0.74	2.17	10.76	32.52	4.05	11.38	5.5x	5.2x
11	Risk Estimate	1.70	1.70	21.00	21.00	27.93	27.93	16.4x	16.4x

^{n}ids of additional properties needed for computation.

cores plus hyper threading) clocked at 3.33GHz and 48GB of main memory. The CPU computations were parallelized using OpenMP, utilizing 24 threads.

In general, it can be seen in Table 1 that using the GPU even for 500 surfaces, the slowest update including skewness and all dependencies plus the probability density function (which needs to be computed for the histogram and time-series views) still allows for interactive update rates. Compared to the CPU version, we achieved a speedup of roughly 5× for all tasks when considering the dependencies.

The histogram, range, mean, variance, kurtosis and the risk estimate are calculated directly from the ensemble and as such the complexity relies solely on the number of surfaces and valid data points per surface. We would expect the computation time for these values to scale linearly with the number of surfaces/valid data points, which seems to be in line with the measured numbers. For larger datasets, however, it would make sense to compute range, mean, variance, kurtosis and the risk estimate using the histogram. This would result in constant time, only depending on the size of the histogram. For the datasets here, however, the histogram computation is the limiting factor. The probability density function, median and mode are looked up using the histogram, and therefore there is no difference between the small and the large data set. Standard deviation and skewness are implemented as simple combinations of other surface properties, and thus computation times are also independent of the number of surfaces. With the dependencies precomputed, the computation of both properties is trivial, which results in very short computation times.

5 Application Scenarios

This section presents two novel visual workflows for planning the placement as well as the operations of off-shore structures (Section 5.1), and for planning underwater glider paths for efficient data acquisition (Section 5.2), respectively. We have designed our framework with these real-world scenarios in mind. During this process, we have worked closely with our domain expert partners, who are also co-authors of this paper, to specifically address their needs. Although we have not yet performed a formal evaluation, we can say that the domain experts are very satisfied with the capabilities provided by our framework, and that they are eager to integrate it with their daily workflow. The close integration of views and tools specifically designed for the tasks at hand, as well as the GPU-based computation and visualization pipeline, considerably speed up the visual analysis process. Our collaborators think that this is a big step forward over the usual approach of combining the actual computation with plotting the results in MATLAB. The biggest advantage of our framework over this more limited approach is provided by the time-series view in our framework. Even though this is a kind of visualization that our partners have not used in their standard workflow until now, they were immediately comfortable with using it in our framework. They thought that the time-series view helps tremendously when judging the forecast for a predefined position over several time samples.

5.1 Off-Shore Oil Operations in the Gulf of Mexico

Oil exploration in the deep Gulf of Mexico is vulnerable to hazards due to strong currents at the fronts of highly non-linear warm-core eddies [51]. The dynamics in the Gulf of Mexico are dominated by the powerful northward Yucatan Current flowing into a semi-enclosed basin. This current forms a loop that exits through the Florida Straits and merges with the Gulf Stream. At irregular intervals, the loop current sheds large eddies that propagate westward across the Gulf of Mexico. This eddy shedding involves a rapid growth of non-linear instabilities [4], and the occasional eddy detachment and reattachment make it very difficult to clearly define, identify, monitor, and forecast an eddy shedding event [3], [15].

The predictability of these eddy shedding events poses a major challenge for the oil and gas industry operating in the Gulf. The presence of these strong currents potentially causes serious problems and safety concerns for the rig operators. Millions of dollars are lost every year due to drilling downtime caused by these powerful currents. As oil production moves further into deeper waters, the costs related to strong current hazards are increasing accordingly, and accurate 3D forecasts of currents are needed. These can help rig operators to avoid some of these losses through better planning.

We illustrate two different scenarios, planning the placement and planning operations of an off-shore platform, with a real-world forecast dataset of the Gulf of Mexico. The dataset covers the Gulf of Mexico basin between $8.5°$ N and $31°$ N, and $72.5°$ W and $98°$ W on a $1/10° \times 1/10°$ grid (225×255 samples of varying metrical distance, between 9.5km and 11km, due to the spherical grid) with 40 vertical layers. Forecasting experiments were performed over a six-month period in 1999 between May and October during which a strong loop current event occurred (Eddy "Juggernaut") [29]. The assimilation cycle was two weeks, resulting in ten temporal samples, each consisting of 50 ensemble members.

Planning Phase. While the accessibility of an existing reservoir is the key factor when planning an oil platform, ocean forecasts can provide valuable additional information. Modern drilling techniques to some extent allow flexible paths and thus considerable flexibility for the actual placement of a platform. However, the complexity of the path has implications on the cost of drilling. On the other hand, slight changes of the position might move a platform from an area that is strongly affected by eddy shedding, which leads to long downtimes, to a less affected area, overall resulting in more efficient operations.

Planning the placement of an off-shore structure requires a complete overview of the ensemble in the spatial domain, but also of all available time steps. Figure 7 outlines all necessary steps. First, the user defines the area of interest (defined by factors not available in the ocean forecast, like reservoir reachability) in the 2D view (Figure 7(a)) for example by painting directly on the map. In Figure 7(b), the sea level of the mean surface for a single sample in time is mapped to the third dimension. The standard deviation is used for pseudo-coloring in the 3D view. By animating over all assimilation cycles, the user can now get an overview of the mean sea level at the selected area of interest, as well as the corresponding uncertainties. Besides the 3D view, animation can also be used in the 2D view, showing the sea level using iso-contours and pseudo-coloring (inset). While the animation is very effective to give a first impression of the changing sea level, it is challenging to derive qualitative results. Therefore, in the next step, the user can look at iso-contours from the mean surfaces, or risk estimates of multiple assimilation cycles in a single view. The contour for a single selected sea level and maximum allowed risk is extracted for all cycles and rendered on the mean surface. The selected sea level, as well as the maximum risk, can be changed on the fly (compare the animation in Figure 7(c)). Starting with a low sea level and zero risk, the user can gradually approach a suitable compromise of available positions, critical sea surface height and risk, to narrow down the area of interest to a few points. Once a compromise is found, the ensemble distribution can be probed interactively at the interesting positions, to verify the results using the histogram view (Figure 7(d)). At this point, the area for placement is narrowed down significantly. Positions for potential placement can now be defined and edited on the map in the 2D view or by entering coordinates directly. The operational phase for these samples can then be simulated identically to the actual operational analysis (see below).

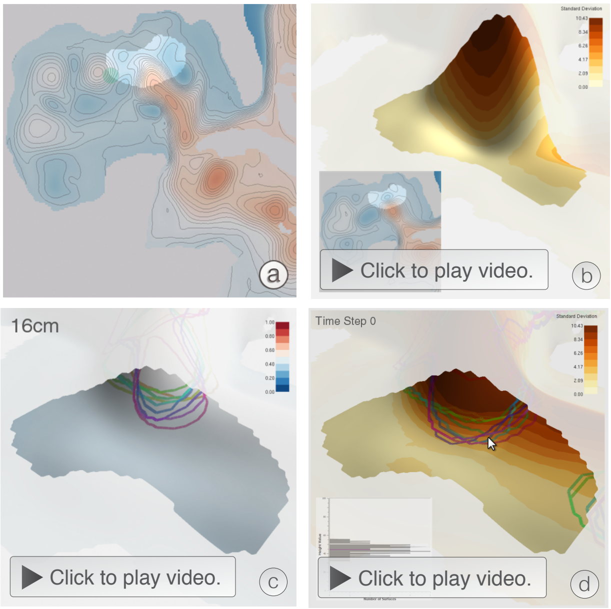
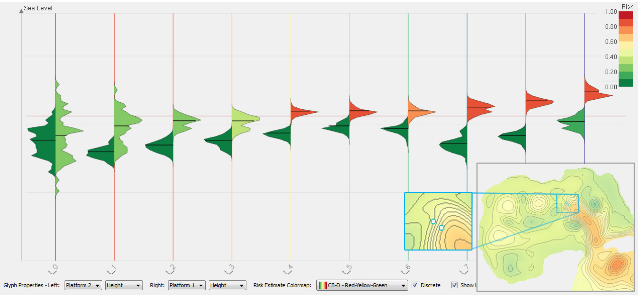


Fig. 7. **Spatial Exploration** for placement planning consists of four main steps: Definition of the area of interest based for example on reservoir reachability (a), general overview (b), time-series analysis (c) and detailed analysis for verification (d).
Please use Adobe Reader $\geq$ 9 to enable animations.



Glyph Properties - Left: Platform 2 Height Right: Platform 1 Height Risk Estimate Colormap: CB-D - Red-Yellow-Green ☑ Discrete ☑ Show L

Fig. 8. **Time-Series Comparison** for two positions (indicated by the markers in the insets) that are in close proximity. The upper left position is mapped to the left side of the glyphs, the lower right position to the right side. The left side of the glyphs shows a position which exhibits very little risk only for the first and last sample of the time-series. Moving the position slightly results in several time steps which would definitely require a shutdown of operations (right side of the glyphs).

Operational Phase. Most of the ensemble analysis for planning operations and unavoidable downtimes is carried out in the time-series view shown in Figure 2(d). For a detailed explanation of the glyph used in the view, see Section 3.2. For planning the operations, the view shows the complete available time-series at a single user-defined spatial position, corresponding to an actual, or potential oil rig. The most important information here is the risk estimate described in Section 3.1. Each glyph is color coded according to the risk at the corresponding point in time. We provide a set of standard color maps for coloring the glyphs. The color map is also freely customizable, most importantly to adapt to the acceptable risk. A good color map should highlight three cases based on the risk estimate: Points in time which are safe for operation with a high certainty, time steps where the rig needs to be shut down with large certainty, and finally uncertain times. We found the green to yellow to red diverging color map, as used in Figure 2(d) to be a good fit, with the green and red mapping to the percentages which indicate safe operations and a high risk, respectively, and the yellow to percentages indicating the need for additional inspection.

The actual operation planning is a recurring process with only a few future assimilation cycles available at a time. Assuming a color map as described, after loading the data the user can immediately identify safe and unsafe points in time from the color of the corresponding glyphs. Only uncertain points in time need further investigation. The main factor to consider for these cases is the spread or uncertainty of the distribution. A compact glyph corresponds to a distribution with little uncertainty. Here, the risk estimate can immediately be used for making a decision to shut down the rig. A large glyph in general indicates large uncertainty. Here, the user must carefully weigh several properties: Are ensemble members in the critical range close to the critical sea level or far above, is the distribution skewed to either side, etc. This information can be derived from the glyph or the user can inspect the raw results from the statistical analysis to make a final decision.

Figure 8 shows a comparison for two positions that are in close proximity, selected for potential placement at the boundary of the eddy shedding area. The spatial distance is less than 30km (compare the positional markers in the insets of Figure 8). While the position mapped to the left side of the glyphs would allow minimal downtimes (only two of the samples of the time-series exhibit any risk, which is also very low) the position mapped to the right side of the glyphs exhibits a very different result. For this position, six out of the ten samples show a sea surface height that will certainly be above the critical level. In addition even the first two samples, which overall are less risky, expose a large amount of uncertainty. Using this comparison one can immediately see that the position mapped to the left side of the glyphs will allow much more efficient operations.

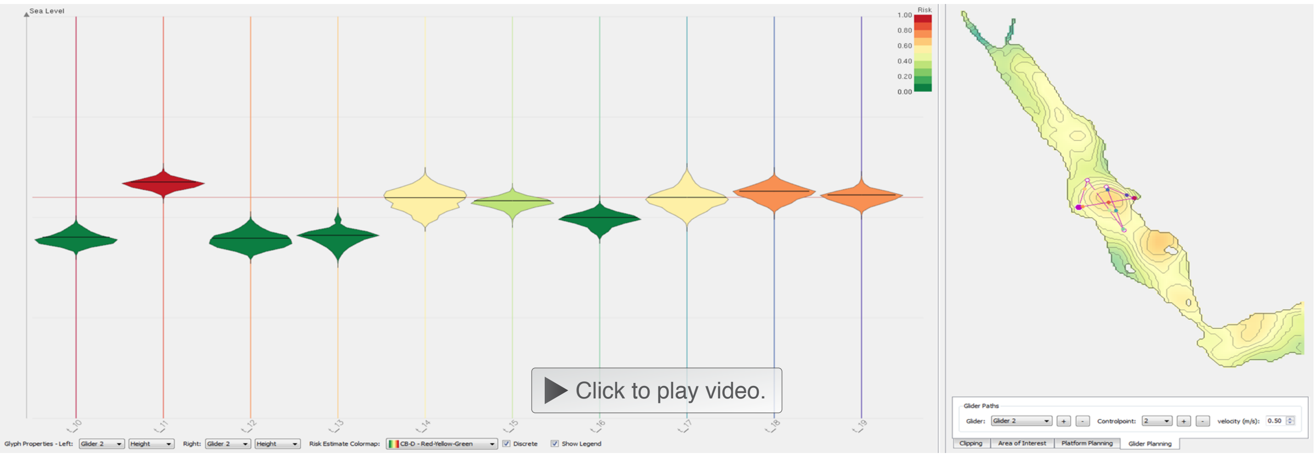
5.2 Planning Glider Paths in the Red Sea

The Red Sea has recently attracted attention from several scientific communities, for its unique physical and biological variability. It is characterized by high temperature and salinity due to its location, surrounded by hot deserts, resulting in high evaporation and negligible precipitation, as well as its isolation from the worlds oceans. Besides the negligible Suez Canal in the north, the only connection to other large bodies of water is through the strait of Bab el Mandeb, a narrow and shallow channel.

The Red Sea is bordered by high mountain ranges that constrain the winds to be closely aligned along the axis of the basin except at a few locations where gaps in the mountains exist. The horizontal circulation in the Red Sea consists of multiple eddies, driven by these strong winds. These eddies strongly influence the exchange of biological organisms and also transport heat and other physical properties. Nowadays, so-called gliders, autonomous underwater vehicles (see Figure 9), are used to record the physical, chemical and biological properties of the eddies, such as temperature, salinity or chlorophyll content.

We illustrate the use of our system for the scenario of planning glider paths using an ensemble simulation dataset for the Red Sea. The dataset covers the Red Sea and extends towards the east through the Gulf of

Fig. 9. **Slocum Glider** as used by the University of Southern California (courtesy of Smith et al. [47]).



Glyph Properties - Left: Glider 2 Height Right: Glider 2 Height Risk Estimate Colormap: CB-D - Red-Yellow-Green ☑ Discrete ☑ Show Legend

Glider Paths
Glider: Glider 2 Controlpoint: 2 velocity (m/s): 0.50

Fig. 10. **Detailed Glider Path Analysis.** The video shows the editing of the path of a glider. When a control point is moved (2D view, right) the sample positions for the time series along the resulting path are recomputed, triggering updates in the time-series view (left) for detailed analysis.
Please use Adobe Reader $\geq$ 9 to enable the video.

Aden to the Arabian Sea and the Persian Gulf. The exact covered area extends from 9° N to 30° N, and 32° E to 77° E on a $1/10° \times 1/10°$ (corresponding to 9.6km to 11km) grid with 50 vertical layers. The months of January to March 2004 were simulated and the assimilation cycle was three days, resulting in a dataset of 30 sampled time steps, with of 50 members, each consisting of 450×210 samples.

Underwater gliders [5], [46], [49] are autonomous sea vehicles consuming very little energy. They move without a propeller, only by means of changing their volume, for example by de- or inflating an external oil bladder, and shifting of weight. Recently Smith et al. [48] proposed improving glider operations by the use of an ocean model and Kalman filtering. The reasons to use ocean forecasts when planning the paths of a glider are diverse. Naturally, the positions of moving eddies are important when one wants to sample data inside these eddies. In addition to this, the energy consumption of the gliders can be judged more precisely when currents along the path are known. Strong currents can also be used as an accelerator, to minimize energy consumption or move the gliders to a desired position more quickly.

In our system, planning the path of a glider is similar to planning the position of an off-shore structure, as described in Section 5.1. First, the user gathers an overview of the eddy positions and their movements over time, using any of the spatial views. Again, the exploration starts with the definition of an area of interest. After that, the user would look at the eddy probability map (or risk estimate), as well as the eddy centers in the 2D and 3D spatial views. In combination with the visualization of the eddy centers, animating over the eddy probability maps of the different samples of the time-series one can easily identify moving and more stationary eddies and plan the path accordingly. The path itself is then defined by placing waypoints in the 2D view. By assigning a velocity to the glider, the positions along the resulting path, corresponding to the available samples of the time-series of the forecast, can be computed. Waypoints can be edited, simply by dragging them within the 2D view. The available positions will automatically be recomputed on the fly. The positions along the path, for which forecast data is available are highlighted in the spatial views, with extra emphasis on the point in time which is currently active in the view. All available positions can be shown in the time-series view. When planning a path, the view behaves differently compared to planning a single fixed position, as described above. While the x-axis still corresponds to the time, each glyph does not only reflect a point in time, but is also created from the data at the grid point along the path computed for this point in time. Figure 10 shows typical adjustments to a path. The user drags one of the waypoints, resulting in variations of the length of the adjacent path segments and thus updates of the positions of the available samples of the time-series along the path. The time-series view updates immediately, showing the detailed information for each of the positions along the path over time.

6 Conclusion

In this work we have presented an interactive, integrated system for the visualization, exploration and analysis of heightfield ensemble data. The core of our framework, which consists of statistical analysis and rendering, is implemented in an efficient GPU-based pipeline. We have illustrated the utility of our framework for two real-world applications based on ocean forecasting. We developed the system in close collaboration with domain expert partners, who now use it on a regular basis. For the future we would like to conduct a formal user study.

In the current state our framework requires the whole dataset to be available in GPU-memory at any point in the pipeline. Even though GPU-memory is getting larger and larger, this is an obvious problem when scaling to very large data. However, the statistics can all be computed based on the histogram. For the future we plan to implement a streaming approach for computing the histogram, i.e. loading the data into GPU memory in slabs and computing the histogram slab by slab. Since the histogram is of constant size, this would eliminate the problem of computing statistics for very large data.

We would also like to explore the possibilities to deploy our framework in a broader set of application scenarios, different from ocean forecasting. While visualization of weather and climate forecasts are obvious targets, completely different areas like analysis of time series of geospatial measurements [50] might also profit from this kind of analysis. In previous work [13] we present the application of our framework for interpretation of seismic tomography data.

ACKNOWLEDGMENTS

The authors wish to thank Burt Jones for his input on glider operations and the anonymous reviewers for their constructive criticism.

REFERENCES

[1] R. A. Brown. Animated visual vibrations as an uncertainty visualisation technique. In *Proceedings of International Conference on Computer Graphics and Interactive Techniques in Australasia and South East Asia*, pages 84–89, 2004.

[2] A. Chaigneau, A. Gizolme, and C. Grados. Mesoscale eddies off peru in altimeter records: Identification algorithms and eddy spatio-temporal patterns. *Progress in Oceanography*, 79(2-4):106–119, 2008.

[3] E. P. Chassignet, H. E. Hurlburt, O. M. Smedstad, C. N. Barron, D. S. Ko, R. C. Rhodes, J. F. Shriver, A. J. Wallcraft, and R. A. Arnone. Assessment of data assimilative ocean models in the gulf of mexico using ocean color. *Circulation in the Gulf of Mexico: Observations and Models*, 161:87–100, 2005.

[4] L. M. Cherubin, W. Sturges, and E. Chassignet. Deep flow variability in the vicinity of the yucatan straits from a high resolution micom simulation. *Journal of Geophysical Research*, 110:20–72, 2005.

[5] C. Eriksen, T. Osse, R. Light, T. Wen, T. Lehman, P. Sabin, J. Ballard, and A. Chiodi. Seaglider: a long-range autonomous underwater vehicle for oceanographic research. *IEEE Journal of Oceanic Engineering*, 26(4):424–436, 2001.

[6] W. W. Esty and J. D. Baneld. The box-percentile plot. *Journal of Statistical Software*, 8, 2003.

[7] G. Evensen. *Data Assimilation: The Ensemble Kalman Filter*. Springer, 2006.

[8] A. E. Gill. *Atmosphere-Ocean Dynamics*. Academic Press, 1982.

[9] H. Griethe and H. Schumann. The visualization of uncertain data: Methods and problems. In *Proceedings of SimVis*, 2006.

[10] G. Grigoryan and P. Rheingans. Point-based probabilistic surfaces to show surface uncertainty. *IEEE Transactions on Visualization and Computer Graphics*, 10(5):564–573, 2004.

[11] C. G. Healey and J. Snoeyink. Vistre: A visualization tool to evaluate errors in terrain representation. In *Proceedings of the Third International Symposium on 3D Data Processing, Visualization, and Transmission*, pages 1056–1063, 2006.

[12] J. L. Hintze and R. D. Nelson. Violin plots: A box plot-density trace synergism. *The American Statistician*, 52(2):181–184, 1998.

[13] T. Höllt, G. Chen, C. D. Hansen, and M. Hadwiger. Extraction and visual analysis of seismic horizon ensembles. *Eurographics 2013 Short Papers*, pages 69–72, 2013.

[14] T. Höllt, A. Magdy, G. Chen, G. Gopalakrishnan, I. Hoteit, C. D. Hansen, and M. Hadwiger. Visual analysis of uncertainties in ocean forecasts for planning and operation of offshore structures. In *Proceedings of the IEEE Pacific Visualization Symposium*, pages 59–66, 2013.

[15] I. Hoteit, T. Hoar, G. Gopalakrishnan, J. Anderson, N. Collins, B. Cornuelle, A. Kohl, and P. Heimbach. A MITgcm/DART ensemble analysis and prediction system with application to the gulf of mexico. *Dynamics of Atmospheres and Oceans*, 63:1–23, 2013.

[16] I. Hoteit, X. Luo, and D.-T. Pham. Particle kalman filtering: A nonlinear bayesian framework for ensemble kalman filters. *Monthly Weather Review*, 140:528–542, 2013.

[17] I. Hoteit, D.-T. Pham, and J. Blum. A simplified reduced order kalman filtering and application to altimetric data assimilation in tropical pacific. *Journal of Marine Systems*, 36:101–127, 2002.

[18] J. A. Jeffrey, T. Hoar, K. Raeder, H. Liu, N. Collins, R. Torn, and A. Avellano. The data assimilation research testbed: A community facility. *Bulletin of the American Meteorological Society*, 90:1283–1296, 2009.

[19] C. R. Johnson and A. R. Sanderson. A next step: Visualizing errors and uncertainty. *IEEE Computer Graphics and Applications*, 23(5):6–10, 2003.

[20] D. Kao, J. Dungan, and A. Pang. Visualizing 2d probability distributions from eos satellite image-derived data sets: a case study. In *Proceedings of the IEEE Visualization Conference*, pages 457–560, 2001.

[21] D. Kao, M. Kramer, A. Love, J. Dungan, and A. Pang. Visualizing distributions from multi-return lidar data to understand forest structure. 42(1):35–47, 2005.

[22] O. D. Lampe and H. Hauser. Curve density estimates. *Computer Graphics Forum*, 30(3):633–642, 2011.

[23] A. Love, A. Pang, and D. Kao. Visualizing spatial multivalue data. *IEEE Computer Graphics and Applications*, 25(3):69–79, 2005.

[24] A. Luo, D. Kao, and A. Pang. Visualizing spatial distribution data sets. In *Proceedings of the Symposium on Data Visualisation*, pages 29–38, 2003.

[25] J. Marshall, A. Adcroft, C. Hill, L. Perelman, and C. Heisey. A finite-volume, incompressible navier stokes model for studies of the ocean on parallel computers. *Journal of Geophysical Research*, 102:5735–5766, 1997.

[26] K. Matkovic, D. Gracanin, B. Klarin, and H. Hauser. Interactive visual analysis of complex scientific data as families of data surfaces. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):1351–1358, 2009.

[27] K. Moreland. Diverging color maps for scientific visualization. In *Proceedings of the 5th International Symposium on Visual Computing*, pages 92–103, 2009.

[28] NVIDIA Corporation. Marching cubes isosurfaces. Available online at http://docs.nvidia.com/cuda/cuda-samples/index.html#graphics (May. 20 2013).

[29] L. Oey, T. Ezer, and H. Lee. Loop current, rings and related circulation in the gulf of mexico: A review of numerical models and future challenges. *Geophysical Monograph-American Geophysical Union*, 161:31, 2005.

[30] A. T. Pang, C. M. Wittenbrink, and S. K. Lodha. Approaches to uncertainty visualization. *The Visual Computer*, 13:370–390, 1997.

[31] T. Pfaffelmoser, M. Reitinger, and R. Westermann. Visualizing the positional and geometrical variability of isosurfaces in uncertain scalar fields. *Computer Graphics Forum*, 30(3):951–960, 2011.

[32] T. Pfaffelmoser and R. Westermann. Visualization of global correlation structures in uncertain 2d scalar fields. *Computer Graphics Forum*, 31(3):1025–1034, 2012.

[33] D. T. Pham. Stochastic methods for sequential data assimilation in strongly nonlinear systems. *Monthly Weather Review*, 129:1194–1207, 2001.

[34] H. Piringer, S. Pajer, W. Berger, and H. Teichmann. Comparative visual analysis of 2d function ensembles. *Computer Graphics Forum*, 31(3):1195–1204, 2012.

[35] K. Pöthkow and H.-C. Hege. Positional uncertainty of isocontours: Condition analysis and probabilistic measures. *IEEE Transactions on Visualization and Computer Graphics*, 17(10):1393–1406, 2011.

[36] K. Pöthkow, B. Weber, and H.-C. Hege. Probabilistic marching cubes. *Computer Graphics Forum*, 30(3):931–940, 2011.

[37] K. Potter, J. Kniss, R. Riesenfeld, and C. R. Johnson. Visualizing summary statistics and uncertainty. *Computer Graphics Forum*, 29(3):823–832, 2010.

[38] K. Potter, A. Wilson, P.-T. Bremer, D. Williams, C. Doutriaux, V. Pascucci, and C. R. Johnson. Ensemble-Vis: A framework for the statistical visualization of ensemble data. In *IEEE Workshop on Knowledge Discovery from Climate Data: Prediction, Extremes and Impacts*, pages 233–240, 2009.

[39] P. J. Rhodes, R. S. Laramee, R. D. Bergeron, and T. M. Sparr. Uncertainty visualization methods in isosurface rendering. In *EUROGRAPHICS 2003 Short Papers*, pages 83–88, 2003.

[40] M. Riveiro. Evaluation of uncertainty visualization techniques for information fusion. In *Proceedings of the 10th International Conference on Information Fusion*, pages 1–8, 2007.

[41] A. Saad, G. Hamarneh, and T. Möller. Exploration and visualization of segmentation uncertainty using shape and appearance prior information. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):1366–1375, 2010.

[42] I. Sadarjoen, F. Post, B. Ma, D. Banks, and H.-G. Pagendarm. Selective visualization of vortices in hydrodynamic flows. In *Proceedings of the IEEE Visualization Conference*, pages 419–422, 1998.

[43] I. A. Sadarjoen and F. H. Post. Detection, quantification, and tracking of vortices using streamline geometry. *Computers & Graphics*, 24(3):333–341, 2000.

[44] J. Sanyal, S. Zhang, J. Dyer, A. Mercer, P. Amburn, and R. J. Moorhead. Noodles: A tool for visualization of numerical weather model ensemble uncertainty. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):1421 – 1430, 2010.

[45] H. Scharsach, M. Hadwiger, A. Neubauer, and K. Bühler. Perspective isosurface and direct volume rendering for virtual endoscopy applications. In *Proceedings of Eurovis*, pages 315–322, 2006.

[46] O. Schofield, J. Kohut, D. Aragon, L. Creed, J. Graver, C. Haldeman, J. Kerfoot, H. Roarty, C. Jones, D. Webb, and S. Glenn. Slocum gliders: Robust and ready: Research articles. *Journal of Field Robotics*, 24(6):473–485, 2007.

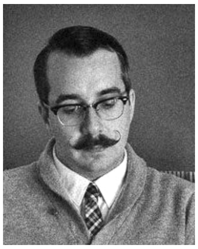
[47] R. N. Smith, J. Kelly, Y. Chao, B. H. Jones, and G. S. Sukhatme. Towards the improvement of autonomous glider navigational accuracy through the use of regional ocean models. In *Proceedings of the 29th ASME International Conference on Ocean, Offshore and Arctic Engineering*, pages 1–10, 2010.

[48] R. N. Smith, J. Kelly, and G. S. Sukhatme. Towards improving mission execution for autonomous gliders with an ocean model and Kalman filter. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 4870–4877, 2012.

[49] H. Stommel. The slocum mission. *Oceanography*, 2(1):22–25, 1989.

[50] S. Thakur, L. Tateosian, H. Mitasova, E. Hardin, and M. Overton. Summary visualizations for coastal spatial-temporal dynamics. *International Journal for Uncertainty Quantification*, 3(3):241–253, 2013.

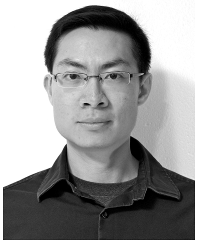
[51] F. M. Vukovich. An updated evaluation of the loop current's eddy-shedding frequency. *Journal of Geophysical Research*, 100:8655–8659, 1995.



Thomas Höllt received the Diplom (MSc) from the University of Koblenz-Landau, Germany, in 2008, and the PhD in computer science from the King Abdullah University of Science and Technology, Saudi Arabia, in 2013. He is a Postdoctoral fellow at the King Abdullah University of Science and Technology. His research interests include scientific visualization, computer graphics and GPGPU. He is a member of ACM and IEEE.

Ahmed Magdy received the Bachelor degree in Computer Science and Engineering from the German University, Cairo, Egypt in 2011 and Masters in Computer Science from the King Abdullah University of Science and Technology, Saudi Arabia in 2012. He is a Software Engineer at Wireless Stars. His research interests include scientific visualization, artificial intelligence, and machine learning.

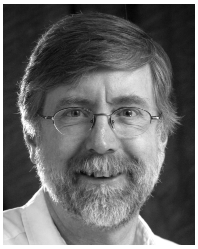
Peng Zhan received the Master's degree in Physical Oceanography from Ocean University of China, and the Master's degree on earth science and engineering from King Abdullah University of Science and Technology. He is currently pursuing his PhD at King Abdullah University of Science and Technology. His research interests include physical oceanography, ocean modeling and assimilation.



Guoning Chen received a PhD degree in computer science from Oregon State University in 2009. He was a post-doctoral research associate at Scientific Computing and Imaging (SCI) Institute at the University of Utah. He is an Assistant Professor at the Department of Computer Science at the University of Houston. His research interests include scientific visualization, computational topology, geometric modeling, and computer graphics. He is a member of ACM and IEEE.

Ganesh Gopalakrishnan received a PhD in Ocean Engineering from the Stevens Institute of Technology in 2009. He is an Assistant Project Scientist at Scripps Institution of Oceanography, University of California San Diego. His research interests include variational and sequential data assimilation, hindcasting, forecasting, and predictability of the ocean state.

Ibrahim Hoteit received a PhD in Applied Mathematics from the University Joseph Fourier, France, in 2002. He is an Assistant Professor with a joint appointment in Earth Sciences and Engineering, and Computer, Electric, and Mathematical Sciences and Engineering at King Abdullah University of Science and Technology. His research interests include data assimilation, ocean modeling, and Bayesian estimation. He has authored more than 50 journal papers.



Charles (Chuck) Hansen received a PhD in computer science from the University of Utah in 1987. He is a Professor of Computer Science and an Associate Director of the Scientific Computing and Imaging Institute at the University of Utah. He has published over 150 peer reviewed journal and conference papers. He was awarded the IEEE Technical Committee on Visualization and Graphics "Technical Achievement Award" in 2005.

Markus Hadwiger received a PhD in computer science from the Vienna University of Technology in 2004. He was a Senior Researcher at the VRVis Research Center in Vienna. He is an Assistant Professor in computer science at King Abdullah University of Science and Technology, Saudi Arabia. He is a co-author of the book Real-Time Volume Graphics. His research interests include petascale visual computing and visualization, and GPU algorithms.