EUROGRAPHICS Symposium on Sketch-Based Interfaces and Modeling (2012)
L. B. Kara and K. Singh (Editors)

Sketch Modeling of Seismic Horizons from Uncertainty

Ronan Amorim[1] Emilio Vital Brazil[1] Daniel Patel[2] Mario Costa Sousa[1]

[1]University of Calgary, Canada [2]University of Bergen, Norway

Abstract

Petroleum reservoir model building is a fundamental but complex task present in all stages of oil/gas exploration and production (E&P). Reservoir models are built incrementally using multi-disciplinary data (e.g. from geophysics, geology, reservoir engineering) and the domain expert interpretation of that data. The first reservoir models are constructed at the appraisal stage, where the available data presents inaccuracies and a high degree of uncertainty. In this paper we present a set of sketch-based interface and modeling operators integrated in a system for the early appraisal stage in oil/gas E&P for the tasks of seismic interpretation and reservoir model building. Our system allows the user to sketch directly over the raw seismic reflection volume and its derived data. These data guide the expert in the key tasks of seismic interpretation and building the structural framework of the reservoir. We propose a novel set of sketch-based modeling operators designed by specific domain requirements from geophysics and geology. A novel architecture using adaptive meshes is also developed to create a more flexible sketch-based system.

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Modeling packages

1. Introduction

A petroleum reservoir, or oil and gas reservoir, is a subsurface accumulation of hydrocarbons contained in porous or fractured rock formations. Most easily identifiable petroleum reservoirs in the subsurface have been discovered. New oil reserves are increasingly more complex and difficult to find. Therefore, improved technology is needed to find new reservoirs and to increase the recovery rate for the existing ones that are in production. Historically, many efforts have been made to improve the prediction accuracy of reservoir behaviors. The accuracy of the prediction is closely related to the quality of the reservoir model being simulated. The first stage of the oil and gas field development is basically to construct the geological 3D structural model. This model will later be used by engineers to simulate and predict the reservoir behavior helping making decisions. Seismic reflection data is used as a volumetric snapshot of the subsurface. The data is of low resolution and often noisy. The separations between rock layers inside the seismic volume are called geological horizons which are a key information to construct the 3D reservoir model. The extraction of this structure is one of the main goals in the seismic analysis process at the early stages of reservoir model building. At this early stage in reservoir model building, the available data (primarily the seismic volume) presents inaccuracies, sparseness, a high degree of uncertainty and they are prone to different interpretations from domain experts [PGT*08, BGSJ07].

A fundamental problem at this appraisal stage is the lack of computational tools to support interactive, intuitive visual interpretation and integration of geophysical data leading to robust conceptual, prototype structural model of the reservoir [PSH*09, GMNH09]. At this early stage in oil/gas field

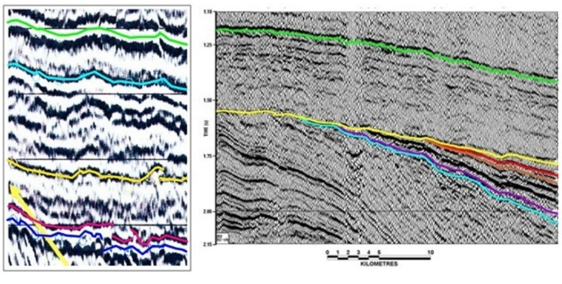

Figure 1: *Manual interpretations of seismic data by means of traces, sketches and annotations. Image courtesy of Primary Industries and Resources, South Australia and Sky Hunter Corporation.*

DOI: 10.2312/SBM/SBM12/001-010

development, modeling the conceptual reservoir framework from seismic data involve various software and numerous manual steps, creating a clumsy intensive workflow. Various important decisions at this early modeling stage require expert domain intepretations where manual intepretations are used by means of sketches, annotations, and hand-drawing traces [PGT*08, PSH*09, Gie09, BGSJ07]. Figure 1 illustrates conventional hand-drawing traces and sketches as part of the seismic interpretation process.

Sketch-based modeling (SBM) tools are a very suitable approach to the problem of how to create, edit or augment geological horizons in the early stages of reservoir modeling. The first version of a reservoir model is constructed at the appraisal stage by interpreting seismic data. Often, the first step is to use automatic or semi-automatic algorithms for extracting horizons [SBMS03,PB05]. These rock boundaries are approximate and come with several topological and geometrical errors. As pointed out by Evans [Eva03], tools are lacking for fast and simple interpretation and creation of reservoir models. Reservoir modeling can be considered a more detailed stage of the interpretation process dealing with accurate positioning of horizons. Reservoir modeling is a domain expert-focused process practiced by only hundreds of specialists worldwide due to non-integrated, complicated software packages having limited concept of model management. This causes a break in productivity and limits interpretation teams to only model the most likely case. As a result, uncertainty around the model is masked and the ability to see the effects of different interpretations on reservoir flow is limited [Eva03]. We attempt to address this problem by offering fast and simple editing of models using sketch-based techniques.

We present a SBM system built based on the petroleum industry requirements and needs for modeling reservoirs at early stages of oil/gas exploration and production (E&P); in addition, this work brings novel computational approaches to SBM issues. The domain was an important motivation for this work, and it guided our designs decisions, but it is not a straightforward implementation of common SBM tools. Indeed, the challenges imposed by the domain guided and inspired us to create a new set of operators (based on experts requirements) as well as a flexible representation structure for these operators. One important requirement of the domain is to be able to control the scales of operations, i.e., local and global manipulations of the model. To allow these different scales of operations we shall use one adaptive mesh [Vel04]. In summary, the main contributions of this work are:

- The use of Sketch-Based Modeling tools to create, augment and edit geological horizons from raw seismic volume data;
- A novel set of SBM operators suitable for geological domain;
- A novel architecture to develop flexible SBM tools using adaptive mesh;

2. Related Work

In this section we present the most related SBM works as well as recent horizon extraction methods.

Igarashi et al. [IMT99] and Nealen et al. [NISA07] developed modeling systems which use curves to create and edit 3D models. Similarly, some of our operators are based on curves but we filter the mouse input using the technique proposed by Vital Brazil et al. [VMC*10]. In contrast with these three systems inflating the model does not make sense because we are working with horizons. Paiva et al. [PAVCS11] present tools to edit and manipulate $4-8$ adaptive meshes. However, the domain obligate us to create a different set of operators. We use parametric surfaces in our system to guide the modeling sections, conversely, Cherlin et al. [CSCSJ05] only used this representation in their SBM system. For a more general classification and taxonomy for SBIM systems consult the work of Olsen et al. [OSCSJ09].

Many works have successfully used SBM techniques to create intuitive work flows and tools for specific tasks and domains. For instance, Anastacio et al. [ASSJ06] proposed a sketch-based interface for modeling plant structures with phyllotactic arrangements. Kara and Shimada [KS07] presented a system for the styling design of 3D objects. The same authors present, one year later [KS08], a method to help car designers to create their final model starting from 2D concepts sketches. Wither et al. [WBCG09] approach a botanic problem of modeling trees using sketches of foliage silhouettes. Lin et al. [LIMS10] create a sketch-based system for the design of sitting posses. Finally, Applegate et al. [ALD11] presented a SBM tool to design realistic road networks for virtual environments. Overall, different domains and specific tasks have had a great impact in the sketch-based research community, they bring motivations and challenges to create new set of intuitive modeling tools.

Although our domain is geological, the application constraints impose us a different set of solutions that have been developed by recent works in terrain sketch-based modeling. These systems usually have as a main target the creation of mountains and valleys [BMV*11,HGA*10,GMS09]. On the other hand, we should follow the seismic volume constraints. So then, some of our operators where designed to allow the expert to use the geological information as a guide.

Most horizon extraction methods are tightly bound to the underlying seismic data due to autotracking (e.g., [FP04]). These automatic methods for extracting horizons from seismic data do not achieve reasonable results due to high noise level and uncertainty. Some semi-automatic methods have been introduced recently. Patel et al. [PBVG10] pre-segment the seismic volume using each voxel as a seed to a growing surface method. The preprocessing is time consuming, but results in that afterwards, the user can interactively choose and assemble pre grown horizon parts for creating horizons. Hölt et al. [HBG*11] propose an interactive system that combines 2D and 3D cost minimization techniques to guide the user through the interpretation horizons process. However, the interactivity is restricted to planes unfolded from prisms created between wells. Engelsma and Hale [EH10] use paint metaphor to select voxels on the seismic volume, this technique allows a level of interactivity and automation. By the same token, changing already interpreted horizons is also not straightforward. To the best of our knowledge this work is the first to present a large number of specific geological SBM operators to edit, augment and create horizons.

3. Seismic Background

Interpreting the subsurface is being performed for finding oil and gas. This is typically done by investigating 3D seismic reflection volumes which are collected by sending sound waves into the ground and analyzing the echoes. The sound waves travel in homogeneous material. When entering a new material with a different impedance, some of the energy is reflected, while the rest continue into the new material. The amount of energy reflected, increases with the impedance difference between the layers. The result is that in seismic reflection volumes, various reflecting layer boundaries of different strength are visible in the seismic data as linear trends. The task of an interpreter is to identify promising layers which may contain oil or gas, based on factors such as the layer shape, findings in neighboring or similar areas, or based on drilled well samples coregistered with the seismic data.

Commercial software for geophysical/geological modeling used by oil companies includes Petrel [Pet] and HydroVR [LLG*07]. In existing tools, the user can extract surfaces by selecting a seed point on a strong reflector. The seed point is then automatically grown into a surface by following the strong reflector through out the volume. These methods are inaccurate, they can give wrong suggestions and can contain holes in noisy areas. In areas where automatic growing fails, the user can draw in horizon lines on vertical equally spaced cross sections to create a coarse grid of the surface. This grid can be fed as input to a growing method to complete it according to the seismic data, or can be given to an interpolation method for creating a minimum energy surface that interpolates the grid independent of the seismic data. The interpreter must use a mix of these methods depending on the quality of the seismic data. Thus, creating a horizon is detail oriented and time consuming. When new data has been collected, such as a well log, an interpreted horizon might need to be altered to match the new data. Fast editing of existing horizons according to new hypotheses is not well supported with existing tools. It is often easier to recreate parts of a horizon that must be changed rather than altering it. We introduce sketch-based modelling and editing of horizons for quickly creating them as an alternative to existing detail oriented and slow methods.

4. System Overview

We have created a flexible architecture to develop sketch-based operators to help the user to model and edit horizons in a rapid and intuitive way (Figure 2). Our system has as input seismic data (reflection volume, distance volume and pre-extracted horizons as shown in Figure 3(a,b,c)), which contains information to guide the experts. The seismic reflection volume data was described in Section 3. The distance volume data and the pre-extracted horizons are pre-computed using the method introduced by Patel et al. [PBVG10]. Details about the input data can be found in Section 4.1. After loading the input data the user can select between different operators designed specifically to meet the experts needs in editing horizons. Each operator has its own types of interaction, including drawing free-form strokes on the actual horizon surface and/or a seismic volume surface.

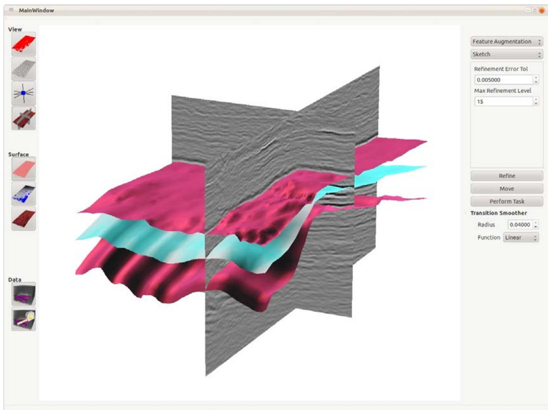

Figure 2: *System interface. Visualization tools on the left, viewer and sketch area in the middle and the operators list and its parameters on the right.*

Due to the locality of some operators and the different resolutions they work, we decided to use an adaptive mesh to permit such different types of interaction. Our choice for the $4-8$ adaptive mesh [Vel04] allows us to introduce a new architecture where the operators are not forced to work with the same surface representation. Instead, each operator can work with its own surface representation, e.g., heightmap, Coons, triangle soup. In addition, these operators can be implemented as separated modules. This approach helps to deal with the restrictions or difficulties of working with a single representation. Details about the adaptive mesh used will be presented in Section 4.2. The operators implemented will be presented in Section 5.

4.1. Workflow

Our system starts modeling horizons from pre-extracted horizons (Figure 3(c)). Even though it is possible to model one or more horizons from scratch. The complete workflow includes the automatically extracted horizons and a distance volume that permits the creation of important modeling and editing operations. Figure 3 presents the complete input dataset.

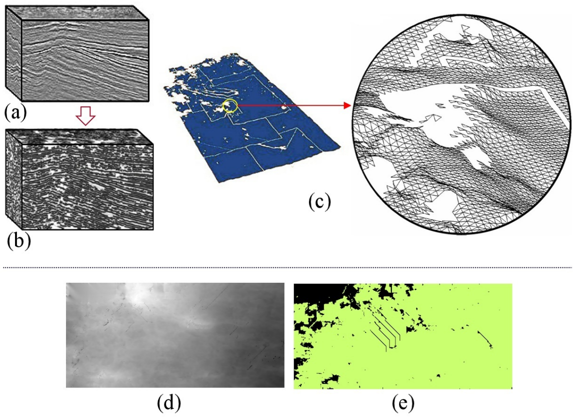

Figure 3: *Input data: (a) seismic volume, (b) distance volume and (c) pre-extracted horizon. Horizon map consisting of (d) height map and (e) hole map.*

The seismic volume (Figure 3(a)) has in each voxel the value of the seismic reflection where lighter gray represents more reflection and indicates the presence of a horizon. The distance volume (Figure 3(b)) is pre-computed using the seismic reflection volume. Each voxel of the seismic reflection volume is used as a seed for a segmentation using a surface growing method [PBVG10]. This method results in a set of horizon patches that are stored in the distance volume. That means, the distance-volume voxel stores the distance to the closest horizon patch segmented. In Figure 3(b) we depicted the distance volume with light gray voxels representing voxels that are farther from a horizon patch than darker voxels.

The pre-extracted horizons (Figure 3(c)) are assembled using the distance volume by combining different horizon patches in an interactive session as described by Patel et al. [PBVG10]. These pre-extracted horizons need to be transformed to a surface representation suitable to our system. This data comes as a triangle soup and is transformed into a *horizon map*, filtering some noise from the original data. This horizon map data is a combination of two different maps: a *height map* and a *hole map* (Figure 3(d,e)). It is important to notice that the input horizons usually come with holes that have to be filled in order to create a height map. The holes are filled using a bilinear interpolation of the closest non-hole parts on the right, left, up and down of the interpolated point. Besides that, a hole map keeps the original holes of the extracted data. We adapt a $4-8$ mesh to the height map and then the horizon is now ready to be edited and augmented using the set of operators.

We propose a simple and flexible architecture. The system is divided in two modules the *main program* and *operators*. The main program is responsible for: managing the horizons ($4-8$ meshes), loading input data, switching the operators, and visualizing. The operators are responsible for modifying the horizons. They have access to input data and also create the policies to change the geometry and/or parameters of the surface. The proposed architecture allows the incorporation of new operators without any changes in other parts of the system. Operators will be described in more details in Section 5.

4.2. Adaptive Mesh

It is very important in the context of this work to use adaptive meshes. Adaptive meshes provide means of having different levels of details within the same mesh. The sketch operations presented in this work are in general local and require different resolutions of the mesh. Usually when we think about a horizon it is a piecewise smooth surface, thus, triangles mesh is an approximation of this surface. Furthermore, there is an error associated with this mesh and a refinement process is done to reduce this error. The use of adaptive meshes avoids the need of a highly refined mesh in areas where the refinement is not necessary. The $4-8$ mesh deals with the adaptive refinement in a transparent way refining the surrounding areas the minimum necessary without the use of different templates and preserving its original structure (Figure 9). Overall, the qualities of the $4-8$ mesh allow us to create a variety of different kinds of operators, each one may have its own surface representation, e.g., height map, parametric surfaces, implicit surfaces, among others.

The $4-8$ mesh always begins from a base mesh that has the desired topology and overall shape needed. This base mesh will be our first approximation, after that, it will be adapted to the desired shape. For our domain application the construction of the base mesh is simple because a plane approximates well the horizon. The adaptation stage is split in two steps. First we move the vertices on the surface and then we refine to decrease the mesh approximation error. Both steps are performed by policies that will be defined by the operators.

We extended the triangle mesh to support different flavors of information. This information allows the operators to perform complex tasks efficiently, moreover, it guarantees the coherence between operators. Each vertex encodes parametric coordinates u and v and the signed distance to the actual sketch, where this distance is calculated in the parametric space. The horizon map is also part of our surface providing means of performing some specific operations.

5. Operators

One of the main challenges creating SBM systems is to define a flexible representation, which allows different forms to manipulate/create the models. Usually, the representation defines the operators and limitations of the system. For instance, using implicit surfaces is hard/expensive to represent details. On the other hand, height maps can represent details but has others limitations, such as the model should be a function of a planar domain. In this work we present a different approach, where each operator can use a different representation, which fits the operation needs. This strategy avoids complications imposed by a single main surface representation.

The operators are an abstract entity to perform changes in the surface, as mentioned in Section 4.2, they implement adaptation to manipulate the $4-8$ meshes. The operators are also registered to receive the input from mouse and keyboard in a filtered way with information of which surface and face (triangle) have been clicked. This information can be used by the operator to implement its sketch metaphors. This system architecture allows the operators' implementation to be a module completely separate from the others. Additionally, the operators can be connected in a hierarchical model. Making it easier to implement different operations starting from basic ones while keeping the whole system manageable and simple. For instance, the *Topology Repair Operator* described in Section 5.1 is an example of connecting two operators. The next sections will describe the *operators* implemented for this work.

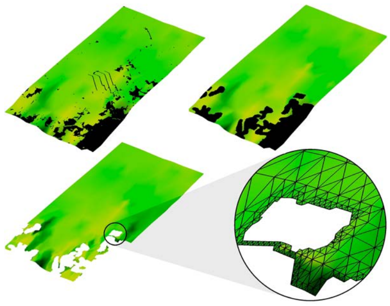

Figure 4: *The user paints on the hole map to delete and add holes to the surface (top row). The surface is then refined and trimmed to the hole map (bottom row).*

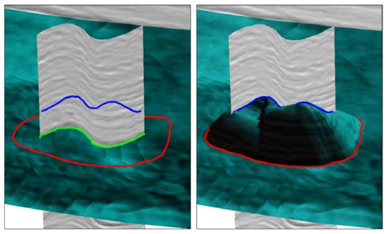

Figure 5: *Feature Augmentation. The sketch in red is the boundary of the deformation area, the green sketch defines the seismic sketch-surface shape and the blue sketch the deformation. On left, all sketches and the seismic sketch surface. On right, the resulting surface.*

5.1. Topology Repair

Automatic and semi-automatic methods for extracting horizons usually result in horizons that have several holes of different sizes and shapes. Most of these holes are result of wrong assumptions of the segmentation algorithm. As described in Section 4.1 we store this topology information in a hole map. The Topology Repair operator allows the user to modify the hole map by drawing on the horizon surface that displays it as a texture. The user can then correct mistakes covering holes or making new holes on the hole map. When the expert is satisfied with the topology, he/she can use it to obtain a horizon with the actual holes. Before cutting the holes the mesh is refined to accommodate the boundary of the holes. Figure 4 presents this process.

This operator illustrates the concept of connecting simple operators to build a complex one. It was implemented as three simple operators: the first one paints the hole map, the second refines the mesh based on the boundaries of the holes, and the third removes the faces under the holes.

5.2. Feature Augmentation

In the traditional work flow of extracting horizons, the expert draws curves on the seismic slices where a horizon would be. Then, these lines set is extrapolated to form a surface. With the same semantic, though more flexible and powerful, our system implements the operator feature augmentation. This operator creates a deformation in the horizon being modeled by the sketch of three curves (Figure 5). The first curve is closed and delimits the region of the surface that will be allowed to be modified. The second curve is inside the delimited region and defines a sketch surface presenting the seismic volumetric data that will guide the expert. It is important to notice that this surface, conversely to the existing tools, defines a free form surface that is not necessarily a plane. This allows the expert to explore all his/her potential to guide the modeling of the horizon. Finally, the user sketches a curve on the seismic sketch surface created defining a deformation of the surface in the region specified by the first sketch.

Given a vertex v_i inside the region we compute its distance d_r to the region boundary and the distance d_s to the seismic surface sketch. The displacement vector $\mathbf{d}$ of v_i is defined by the difference of the projection of v_i on the seismic surface sketch $P_s(u)$ to its corresponding point on the deformation sketch $P_d(u)$, i.e., $\mathbf{d} = P_d(u) - P_s(u)$. The new position of v_i, is found by $v_i + \alpha\mathbf{d}$, where $0 \leq \alpha \leq 1$. The value of α is computed as a function d_r and d_s, for instance $\alpha = 1 - \frac{d_s}{d_r + d_s}$.

5.3. Horizon Fault Deformation

Horizons are usually discontinuous surfaces, these discontinuities are called faults. Faults are fractures in the volume of rocks where there is a significant displacement along the fractures. Creating a fault is a very important step in the horizon interpretation work flow. This operator creates a fault cutting a sketched curve on the surface and then displacing one of the sides. We use three sketches to define a fault (Figure 6). The first sketch defines where the horizon surface will be cut to create the fault. A second sketch defines a region of interest and the third sketch is performed on a seismic surface that follows the first sketch, and defines how the fault will be displaced. The vertex displacement is performed in a similar way as described in Section 5.2. Where vertices with zero distances to the fault sketch will be moved to the displacement sketch and vertices with zero distances to the region delimiting sketch will not be displaced.

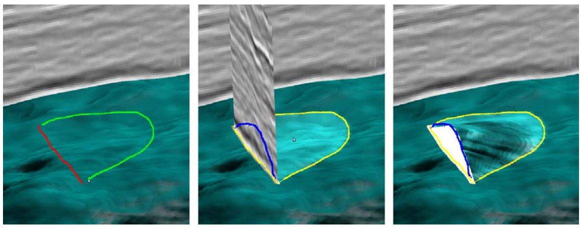

Figure 6: *Creating a deformed fault in the horizon surface. The user sketches the cut, red line, and the region of interest, green line (left). The seismic volume is projected in the fault surface and the user draws the displacement, blue sketch (middle). Operator results (right).*

5.4. Magnetic

In the existing horizon extracting tools the user can choose one voxel to be used as a seed in a growing segmentation algorithm, resulting in a horizon patch. This method is usually used in the editing stage of the work flow. The assembly process of different surfaces patches is usually cumbersome in the traditional pipeline. The magnetic operator, has the semantic of instantaneously growing many seed voxels at the same time, in contrast to the traditional work flow. The user approximates the surface using other operators and then snaps it the pre-extracted horizons. This operator is guided by the distance volume, which has the distances in each voxel to the closest patch, previously segmented. Thus, each vertex of the mesh is displaced to the local minimum in its z-coordinate. This operator can be applied to a whole horizon surface or to a selected area. Figure 7 presents the magnetic operator applied to a sketched region.

Another example of usage of this operator is snapping a surface patch, instead of directly snapping the actual surface. This surface patch could either be extracted from the actual surface or created from scratch. After that, this surface patch is joined to the actual horizon (Figure 8).

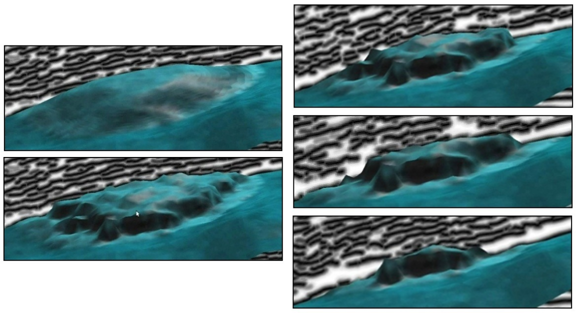

Figure 7: *Applying the magnetic operator (left) and moving the seismic plane to inspect the fitting with the distance field (right).*

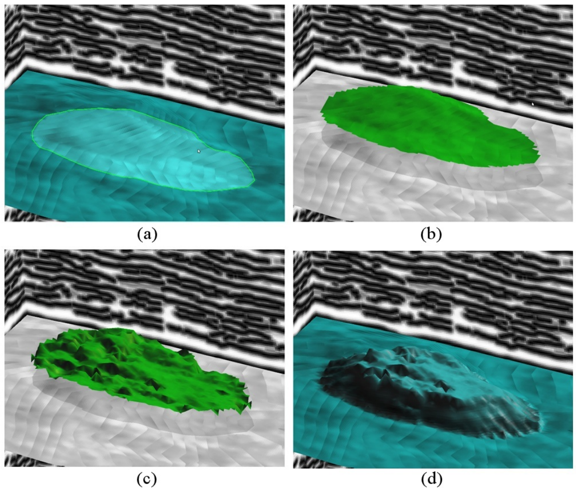

Figure 8: *Using a different horizon patch to fit the magnetic operator and then joining with the main horizon. Region is first interactively selected (a) and raised (b); magnetic operator is applied (c) and then the main horizon surface is snapped to the surface patch (d).*

5.5. Horizon Convex Sum

A horizon is often approximated well by a convex sum of the horizons above and below it. Based on this fact we designed the Horizon Convex Sum operator that creates a new horizon using two others. A new surface S_n is created as a convex sum of the two others, S_1 and S_2. That means, a point $P \in S_n$ is calculated as $\alpha P_1 + (1-\alpha)P_2$, where $P_1 \in S_1$, $P_2 \in S_2$ and $0 \leq \alpha \leq 1$ chosen by the user. These points are related by their u and v coordinates (Figure 16(e)).

5.6. Supporting Operators

Some other operators were implemented to provide modeling tools which are not necessarily specific to our application domain. Next we describe these operators.

The **sketch operator** was implemented to define regions of interest. Many of the operators define a region of interest by a closed curve on the surface. To have a region of interest well defined it is necessary to have edges of the $4-8$ mesh matching the sketch. Usually this is not true, so it is necessary to refine the mesh around the sketch to approximate this behavior. Our sketch is filtered by the method described by Vital Brazil et al. [VMC*10] which consists of sampling the sketch evenly and then applying a reverse Chaikin subdivision. The sketch is then projected on the parametric coordinates of our surface where it will allow us to create simple rules to refine the mesh. This operator implements the rules to refine the mesh by refining the edges that have an intersection with the sketch. This process is repeated until the edge reaches a pre-defined maximum refinement level or the distance between its vertices to the sketch is lower than an error. Figure 9 presents the adaptive refinement of the $4-8$ mesh to a sketch.

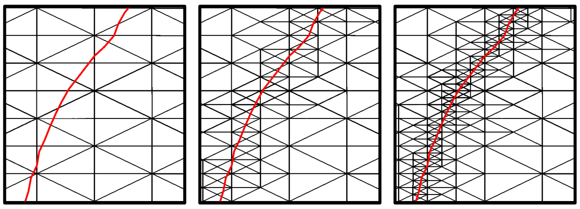

Figure 9: *Sketch refinement. Sketch over the original mesh on left, and two approximations for different error values.*

The **pinch operator** was implemented to allow some rapid adjustments of the horizon surface. In this operator the user pinches a point on the horizon surface and drags it to any direction. This operator has two parameters: a radius of influence and a decay function. The radius defines the region of interest while the decay function gives the overall shape. The decay function has values between 0 and 1 and scales the displacements. Figure 10 presents the pinch method using 2 different decay functions.

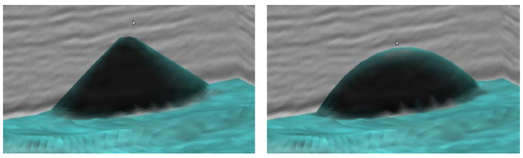

Figure 10: *On left, the pinch performed using linear decay function of the distance to displace vertices. On right, the pinch performed using quadratic decay function of the distance to displace vertices.*

The **move operator** was implemented as a generalization of the pinch operator. It moves a region of interest, sketched by the user instead of only a single point. The radius (here defined as a distance to the sketched region) and the region of interest are used to calculate decay function. Outside the region of interest the decay function is zero and inside, if the distance is less than the radius we apply the decay function, otherwise it is one. Figure 11 presents examples of the move operator with and without the use of a decay function.

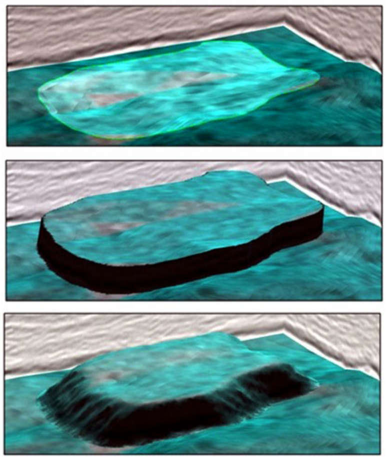

Figure 11: *Moving up a region of interest sketched by the user (top). Moving without decay function (middle). Moving with decay function (bottom).*

The **smooth operator** was implemented to smooth surfaces areas. Due to the use of segmentation algorithms in the input data, and other modeling problems, the horizon surface can contain some noisy areas. Such areas can be smoothed by a *smooth operator* that displaces a vertex to the new z position computed as the average position of its star (Figure 12).

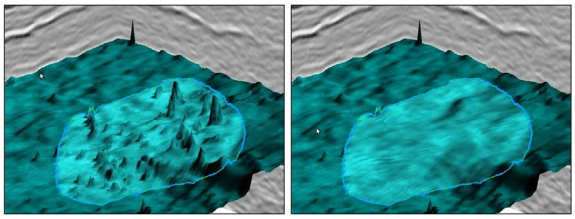

Figure 12: *Smooth operator. On left, the user sketches a region. On right, the selected region is smoothed.*

The **local surface snapping operator** has source and target surfaces. The target surface is deformed to snap the source surface. This particular operator takes advantage of the parametric information of both surfaces to displace vertices of the target surface to the corresponding parametric coordinate in the source surface. The target surface is refined in areas where the details of the other surface require so. The user has the option to use a decay function in a radius to have a smooth result (Figure 8 (c,d)). The **coons surface operator** was implemented to allow the user to create horizons from scratch by drawing four curves that define a coons surface. Figure 16(b, c) present this operator and a horizon created.

The **surface extractor operator** duplicates one region of a horizon as a new surface patch. The region is defined by a closed curve sketched by the user (Figure 8).

6. Results and Discussions

In this section we present three case studies of usage of our system combining different operators. We also present preliminary user feedbacks and technical aspects of the implementation. Finally, we discuss some limitations of our system.

In our first case study (Figure 13) we deform a pre-extracted horizon in an area with potential problem. A region of interest is sketched and is moved up guided by the distance volume (a,b). The magnetic operator is then applied to the same region to fit to the horizon candidates nearby (b). The magnetic operator yields a noisy surface that we apply the smooth operator to have a more regular surface (b). The surface is then freely edited in areas around the previous sketch by the pinch operator (c). Since we have the seismic reflection as a 3D texture in the horizon the user uses it to guide the pinch. Then the magnetic operator is applied in conjunction with the smooth operator again to fit the pinched areas (d). The user then uses the feature augmentation operator to perform the final horizon adjustments (e,f).

In the second case study, in the first stage (Figure 14), a region is sketched and a surface patch is extracted from there. The surface patch is then moved to a desired region and snapped to a candidate horizon with the magnetic operator (b). The smooth operator is applied resulting in a more regular surface (c). The surface patch is now moved back to level of the surface it was extracted from (d) and then snapped, copying characteristics of the horizon candidate above the horizon being modeled (e). In the second stage (Figure 15), a fault is created and deformed based on the seismic data (a). Finally, the repair operator is applied and the hole map is modified to fix wrong holes of the pre-extracted horizon (b ,c). The horizon is then trimmed by the new hole map (c,d).

In our final case study (Figure 16) we start from one well-finished horizon (a). Then, a new horizon is created from scratch by drawing four curves using as a guide the distance volume slices (b, c). A region of the newly created horizon is adjusted by sketching a region that is moved up and then snapped to a candidate horizon using the magnetic operator (d). Finally, a new horizon is created as a convex sum of the two existing horizons and then fitted with the magnetic operator (e).

Our system is still in process of evaluation by collaborators in the geological domain. Their evaluation so far have been positive, with some suggestions of new operators and adjustments. The magnetic operator was well received when used in conjunction of the smooth operator. The horizon fault deformation operator was observed to be very useful because one of the main drawbacks of the actual horizon extraction pipeline is to model faults. But some suggestions were made to improve the system and operators, including allowing more flexible ways of drawing faults and a better navigation (compass and scale). Our system also has other limitations such as the lack of visual feedback for operators like pinch and repair, where a region of influence could guide the user better. The fault deformation operator would benefit of an edge snapping operator allowing a more precise cut of the surface. Other operators could also benefit of the same edge snapping although it is not a perfect solution because the snapping can modify the geometry in regions where the user does not want to change.

The prototype of our system was implemented to test different operators and how they can be combined to perform more complex tasks. The system was implemented using the C++ programming language, the visual interface was built using Qt and we used the $4-8$ adaptive mesh library described by Velho [Vel04] and available in [MV11]. All case studies were generated on a 2.66 GHz Intel Xeon W3520, 4GB RAM and OpenGL/nVidia GeForce GTX470 Graphics. All operators achieved interactive time. The most expensive operation is deleting faces (used in topology repair and horizon fault deformation operators) depending on the number of faces being deleted it could take about 4 seconds (deleting 2.5k triangles from a 70k triangles mesh). The reason is the need to rebuilt the mesh connectivity. Finally, to speed up the evaluation of surface properties given (u,v) coordinates we implement a simple quadtree.

7. Conclusion and Future Work

In this work we presented a SBM system to model geological horizons. This system provides a new set of operators designed with the geological domain in mind to help the expert to edit, model and augment horizons. To develop the system we presented a novel architecture to create flexible SBM tools using an adaptive mesh. The SBM tools developed have shown promising results in terms of allowing a more intuitive and rapid modeling of horizons.

The tools we presented in this paper are part of a first prototype under joint research and development with partners from the oil and gas industry. As future work we plan to extend our system to have more operators and formal system and usability evaluations following guidelines provided by experts feedbacks and requirements from specific reservoir modeling and geological scenario studies.

Acknowledgments

We would like to thank our colleagues for their useful discussions and advice, in particular to Felipe Albuquerque Portella and Camila Faria de Albuquerque from Petrobras SA. Many thanks

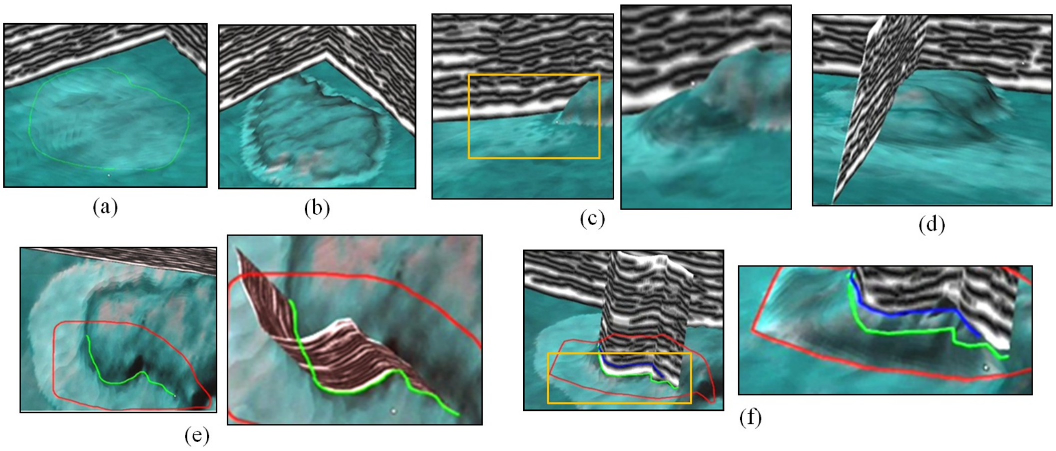

Figure 13: *Screen shots of the first case study.*

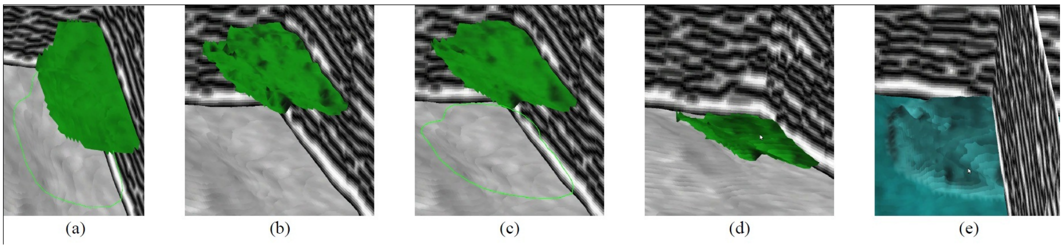

Figure 14: *Screen shots of the second case study.*

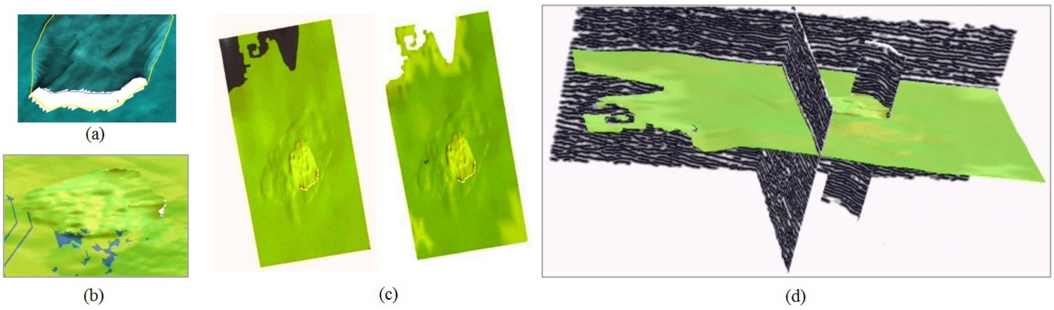

Figure 15: *Screen shots of the second case study during the stage of fault modeling and topology fixing.*

to Primary Industries and Resources, South Australia, and to Russ Duncan, Ken Bradley, and Aaron Balasch from Sky Hunter Corporation. We also thank the anonymous reviewers for their careful and valuable comments and suggestions. This research was supported in part by the NSERC / Alberta Innovates Technology Futures (AITF) / Foundation CMG Industrial Research Chair program in Scalable Reservoir Visualization. Parts of this work have been carried out within the Geoillustrator research project, which is funded by Statoil and the PETROMAKS programme of The Research Council of Norway. Data used in this article is by courtesy of Norske ConocoPhillips AS, ExxonMobil Norway AS, Total E&P Norge AS, Statoil AS and Petoro AS.

References

[ALD11] APPLEGATE C. S., LAYCOCK S. D., DAY A. M.: A sketch-based system for highway design. In *SBIM '11* (New York, NY, USA, 2011), ACM, pp. 55–62. 2

[ASSJ06] ANASTACIO F., SOUSA M. C., SAMAVATI F., JORGE J. A.: Modeling plant structures using concept sketches. In *NPAR '06* (2006), ACM, pp. 105–113. 2

[BGSJ07] BOND C., GIBBS A., SHIPTON Z., JONES S.: What do you think this is?: "conceptual uncertainty" in geoscience interpretation. *GSA Today 17*, 11 (2007), 4–10. 1, 2

[BMV*11] BERNHARDT A., MAXIMO A., VELHO L., HNAIDI H., CANI M.-P.: Real-time terrain modeling using cpu-gpu coupled computation. In *SIBGRAPI '11* (2011), pp. 64–71. 2

[CSCSJ05] CHERLIN J. J., SAMAVATI F., COSTA SOUSA M., JORGE J. A.: Sketch-based modeling with few strokes. In *SCCG '05* (2005), ACM, pp. 137–145. 2

[EH10] ENGELSMA C., HALE D.: Painting seismic images in 3d. *SEG Technical Program Expanded Abstracts 29*, 1 (2010), 1271–1275. `doi:10.1190/1.3513075`. 3

[Eva03] EVANS S.: The last integration frontier. *E&P February*

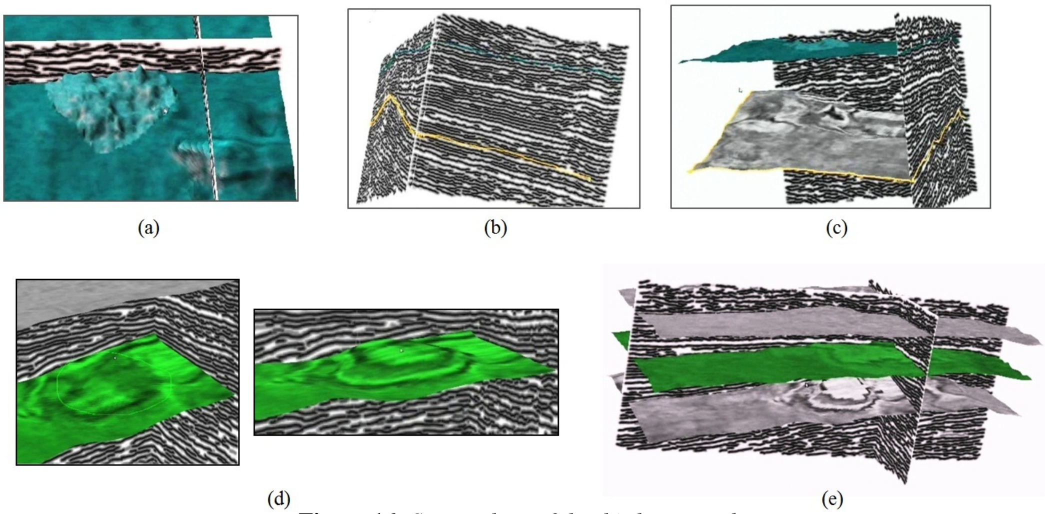

Figure 16: *Screen shots of the third case study.*

(2003). URL: http://www.epmag.com/EP-Magazine/2003/02/. 2

[FP04] FARAKLIOTI M., PETROU M.: Horizon picking in 3d seismic data volumes. *Mach. Vision Appl. 15*, 4 (Oct. 2004), 216–219. 2

[Gie09] GIERTSEN C.: Applications of illustrative methods for oil & gas exploration and production. In *First Interdisciplinary Gathering on Illustrative Visualization (IllustraVis '09), University of Bergen, Norway.* (2009). 2

[GMNH09] GEIGERN S., MATTHÃĎI S., NIESSNER J., HELMIG R.: Black-oil simulations for three-component, three-phase flow in fractured porous media. *SPE Journal 14*, 2 (2009), 338–354. 1

[GMS09] GAIN J., MARAIS P., STRASSER W.: Terrain sketching. In *I3D '09* (2009), ACM, pp. 31–38. 2

[HBG*11] HÖLLT T., BEYER J., GSCHWANTNER F., MUIGG P., DOLEISCH H., HEINEMANN G., HADWIGER M.: Interactive seismic interpretation with piecewise global energy minimization. In *PacificVis'11* (2011), pp. 59 –66. 3

[HGA*10] HNAIDI H., GUÉRIN E., AKKOUCHE S., PEYTAVIE A., GALIN E.: Feature based terrain generation using diffusion equation. *Comp.r Graph. Forum (Proceedings of Pacific Graphics) 29*, 7 (2010), 2179–2186. 2

[IMT99] IGARASHI T., MATSUOKA S., TANAKA H.: Teddy: a sketching interface for 3D freeform design. In *Proc. of SIGGRAPH '99* (1999), ACM, pp. 409–416. 2

[KS07] KARA L. B., SHIMADA K.: Sketch-based 3D-shape creation for industrial styling design. *IEEE Comput. Graph. Appl. 27*, 1 (2007), 60–71. 2

[KS08] KARA L. B., SHIMADA K.: Supporting early styling design of automobiles using sketch-based 3D shape construction. *Computer-Aided Design and App. 5*, 6 (2008), 867–876. 2

[LIMS10] LIN J., IGARASHI T., MITANI J., SAUL G.: A sketching interface for sitting-pose design. In *SBIM '10* (2010), pp. 111–118. 2

[LLG*07] LIDAL E. M., LANGELAND T., GIERTSEN C., GRIMSGAARD J., HELLAND R.: A decade of increased oil recovery in virtual reality. *IEEE Computer Graphics and Applications 27*, 6 (2007), 94–97. 3

[MV11] MAXIMO A., VELHO L.: a48 2.0.2 library, 2011. last visit: Mar. 2012. URL: http://w3.impa.br/~andmax/a48/. 8

[NISA07] NEALEN A., IGARASHI T., SORKINE O., ALEXA M.: Fibermesh: designing freeform surfaces with 3D curves. *ACM Trans. Graph. 26*, 3 (2007), 41–50. 2

[OSCSJ09] OLSEN L., SAMAVATI F. F., COSTA SOUSA M., JORGE J.: Sketch-based modeling: a survey. *Computer & Graphics 33*, 1 (2009), 85–103. 2

[PAVCS11] PAIVA A., AMORIM R., VELHO L., COSTA SOUSA M.: Sketch-based adaptive mesh augmentation using stellar operators. In *SIBGRAPI'11* (2011), pp. 78–85. 2

[PB05] PEPPER R., BEJARANO G.: Advances in seismic fault interpretation automation. *Search and Discovery* (2005), Article #40169. URL: http://www.searchanddiscovery.com/documents/2005/pepper/. 2

[PBVG10] PATEL D., BRUCKNER S., VIOLA I., GROLLER E.: Seismic volume visualization for horizon extraction. In *PacificVis'10* (2010), IEEE, pp. 73–80. 2, 3, 4

[Pet] PETREL S.: Petrel seismic interpretation software. Schlumberger Information Solutions (SIS). 3

[PGT*08] PATEL D., GIERTSEN C., THURMOND J., GJELBERG J., GRÖLLER E.: The seismic analyzer: Interpreting and illustrating 2d seismic data. *IEEE Trans. on Vis. and Comp. Graph. 14*, 6 (2008), 1571–1578. 1, 2

[PSH*09] PATEL D., STURE O., HAUSER H., GIERTSEN C., GRÖLLER M.: Knowledge-assisted visualization of seismic data. *Comp. & Graph. 33*, 5 (2009), 585–596. 1, 2

[SBMS03] SHEFFIELD T. M., BULLOCH T. E., MEYER D., SUTTON J.: Geovolume visualization and interpretation: Speed and accuracy with auto-tracking. In *2003 SEG Annual Meeting* (2003), Society of Exploration Geophysicists. 2

[Vel04] VELHO L.: A dynamic adaptive mesh library based on stellar operators. *Journal of Graphics Tools 9*, 2 (2004), 21–47. 2, 3, 8

[VMC*10] VITAL BRAZIL E., MACÊDO I., COSTA SOUSA M., DE FIGUEIREDO L. H., VELHO L.: Sketching variational Hermite-RBF implicits. In *SBIM '10* (2010), pp. 1–8. 2, 7

[WBCG09] WITHER J., BOUDON F., CANI M.-P., GODIN C.: Structure from silhouettes: a new paradigm for fast sketch-based design of trees. *Comp. Graph. Forum 28*, 2 (2009), 541–550. 2