Sketch-based Modelling and Conceptual Visualization of Geomorphological Processes for Interactive Scientific Communication

Mattia Natali

Dissertation for the degree of Philosophiae Doctor (PhD)

Supervised by Dr. Daniel Patel
Co-supervised by Prof. Ivan Viola

Department of Informatics
University of Bergen
Norway

May 2014

Scientific Environment

THE research work of this dissertation has been carried out in the Visualization Group at the Department of Informatics, Faculty of Mathematics and Natural Sciences, University of Bergen (UiB), Norway. The work has been part of the interdisciplinary research project called *GeoIllustrator*, in collaboration with the Department of Earth Science and Christian Michelsen Research (CMR). The project was funded by Statoil and the Norwegian Research Council under the Petromaks programme (#200512). One of my two co-supervisors is professor William Helland-Hansen from the Department of Earth Science (UiB).

UNIVERSITY OF BERGEN
Department of Informatics

UNIVERSITY OF BERGEN
Faculty of Mathematics and Natural Sciences

GEO ILLUSTRATOR

cmr Computing

Acknowledgements

I would like to thank my advisors Daniel Patel and Ivan Viola, who have helped me during the entire process that has led to this thesis. I also thank professor Helwig Hauser, who has always been available for suggestions on important decisions even if he has not officially been one of my supervisors.

I appreciate the feedback offered by professor William Helland-Hansen and Tore Grane Klausen, since they have provided a geological point of view into my research.

I am grateful to present and past members, during my PhD studies, of the visualization group in Bergen. Particularly to Endre Mølster Lidal, who has collaborated with me in the PhD project and shared his experience.

I have also benefited from the help of the administrative staff at the department.

Besides the university environment, I would like to express my gratitude to all the people that have shared a piece of my life experience in Norway, each of them on different aspects, interests and activities.

My acknowledgements go to friends and family at home too. They have supported me during my PhD journey.

Abstract

Throughout this dissertation, solutions for rapid digitalization of ideas will be defined. More precisely, the focus is on interactive scientific sketching and communication of geology, where the result is a digital illustrative 3D model. Results are achieved through a sketch-based modelling approach which gives the user a more natural and intuitive modelling process, hence leading to a quicker definition of a geological illustration.

To be able to quickly externalize and communicate ones ideas as a digital 3D model, can be of importance. For instance, students may profit from explanations supported by interactive illustrations. Exchange of information and hypotheses between domain experts is also a targeted situation in our work. Furthermore, illustrative models are frequently employed in business, when decisional meetings take place for convincing the management that a project is worth to be funded.

An advantage of digital models is that they can be saved and they are easy to distribute. In contrast to 2D images or paper sketches, one can interact with digital 3D models, and they can be transferred on portable devices for easy access (for instance during geological field studies). Another advantage, compared to standard geological illustrations, is that if a model has been created with internal structures, it can be arbitrarily cut and inspected.

Different solutions for different aspects of subsurface geology are presented in this dissertation. To express folding and faulting processes, a first modelling approach based on cross-sectional sketches is introduced. User defined textures can be associated to each layer, and can then be deformed with sketch strokes, for communicating layer properties such as rock type and grain size.

A following contribution includes a simple and compact representation to model and visualize 3D stratigraphic models. With this representation, erosion and deposition of fluvial systems are easy to specify and display. Ancient river channels and other geological features, which are present in the subsurface, can be accessed by means of a volumetric representation.

Geological models are obtained and visualized by sequentially defining stratigraphic layers, where each layer represents a unique erosion or deposition event. Evolution of rivers and deltas is important for geologists when interpreting the stratigraphy of the subsurface, in particular because it changes the landscape morphology and because river deposits are potential hydrocarbon reservoirs.

Time plays a fundamental role in geological processes. Animations are well suited for communicating temporal change and a contribution in this direction is also given.

With the techniques developed in this thesis, it becomes possible to produce a range of geological scenarios. The focus is on enabling geologists to create their subsurface models by means of sketches, to quickly communicate concepts and ideas rather than detailed information.

Although the proposed techniques are simple to use and require little design effort,

complex models can be realized.

Publications

This thesis is based on the following publications:

Paper A: **Mattia Natali**, Ivan Viola and Daniel Patel. Rapid Visualization of Geological Concepts. In *Proceedings of SIBGRAPI Conference on Graphics, Patterns and Images*, 2012, pp. 150–157.

Paper B: **Mattia Natali**, Tore Grane Klausen and Daniel Patel. Sketch-Based Modelling and Visualization of Geological Deposition. In *Computers & Geosciences 67C*, 2014, pp. 40–48.

Paper C: **Mattia Natali**, Julius Parulek and Daniel Patel. Rapid Modelling of Interactive Geological Illustrations with Faults and Compaction. In proceedings of *Spring Conference on Computer Graphics (SCCG)*, 2014.

The following publications are also related to the dissertation:

Paper 1: **Mattia Natali**, Endre Mølster Lidal, Julius Parulek, Ivan Viola, Daniel Patel. Modeling Terrains and Subsurface Geology. In *Proceedings of Eurographics 2013 (STARs)*.

Paper 2: Endre Mølster Lidal, **Mattia Natali**, Daniel Patel, Helwig Hauser, Ivan Viola. Geological Storytelling. In *Computers & Graphics 37*, 5 (2013), pp. 445–459.

Moreover, the following articles on non-related topics were published during my Ph.D. studies:

Paper I: **Mattia Natali**, Silvia Biasotti, Giuseppe Patané, Bianca Falcidieno. Graph-based Representations of Point Clouds. In *Graphical Models 73*, 5 (2011), pp. 151–164.

Paper II: **Mattia Natali**, Marco Attene, Giulio Ottonello. Steepest Descent Paths on Simplicial Meshes of Arbitrary Dimensions. In *Computers & Graphics 37*, 6 (2013), pp. 687–696.

I am the main author of all the papers listed above except for Paper 2, where I contributed with implementation and a description of the automatic generation of 3D illustrative animation from 2D animated sketches. Paper 2 was co-authored with Endre Mølster Lidal, who was my fellow PhD student in the GeoIllustrator project at the time the paper was written.

My thesis is based on Paper A, Paper B and Paper C. The research conducted to produce these three papers was guided by both Daniel Patel and professor Ivan Viola,

respectively my main supervisor and my co-supervisor. They also contributed to the writing of Paper 1 and Paper 2.

Tore Grane Klausen, who received his PhD in geology in 2013 as part of the GeoIllustrator project, contributed in Paper B with regard to the geoscientific domain motivation of our research. He also provided valuable feedback for developing the prototype we used to create our results.

Julius Parulek co-authored Paper C and Paper 1. In Paper C, he contributed with his expertise on GPU programming and wrote parts of the text describing this.

Thesis Structure

The thesis consists of two parts. Part I gives an introduction to the problems that are addressed in the thesis, a short overview of related work, and summarizes the main contributions we achieved in five chapters. Chapter 1 introduces our research topic and our main contributions. We give an overview of related work in geological modelling in Chapter 2. In Chapter 3, we introduce the methods we developed in our scientific work, which are detailed in the papers of Part II. Results and demonstration cases are shown in Chapter 4, while conclusions and future work are stated in Chapter 5.

Part II presents the three papers that form the basis of this dissertation, reformatted and with a common bibliography.

Contents

Scientific Environment iii

Acknowledgements v

Abstract vii

Publications ix

Thesis Structure xi

I Overview 1

1 Introduction 3
1.1 Problem Statement . . . 3
1.2 Requirement Analysis . . . 5

2 Modelling and Visualization in Geology - State of the Art 7
2.1 Geomodelling in Computer Graphics and in Geosciences . . . 8
2.2 Geological Objects . . . 8
2.3 Modelling of Geological Objects . . . 9
2.4 Geomodelling Data Taxonomy . . . 10
2.4.1 Data-Free Scenario . . . 11
2.4.2 Sparse and Dense Data Scenarios . . . 17
2.5 Illustrative Visualization . . . 21
2.6 Challenges and trends in geological modelling . . . 21

3 Modelling and Visualization - Extending the State of the Art 23
3.1 Contributions . . . 24
3.2 Rapid Modelling . . . 24
3.3 Illustrative Animated Storytelling . . . 26
3.4 Layered Data Representation for Morphological Evolution . . . 27
3.5 Model Discontinuities and Interactive Illustrations . . . 28

4 Results 31
4.1 Folds, Faults and Erosion . . . 31
4.2 Stratification and Fluvial Systems . . . 33
4.3 Interactive Faults and Compaction . . . 38

5 Conclusions and Future Work 41

II Scientific Results 43

A Rapid Visualization of Geological Concepts 45
A.1 Introduction . . . 46
A.2 Motivation . . . 48
A.3 Related Work . . . 48
A.4 Methodology . . . 51
A.4.1 Folding . . . 51
A.4.2 Faulting . . . 52
A.4.3 Texturing . . . 53
A.4.4 Projected Drawing . . . 57
A.5 Results . . . 57
A.6 Conclusions . . . 59
A.7 Future Work . . . 60

B Sketch-Based Modelling and Visualization of Geological Deposition 61
B.1 Introduction . . . 62
B.2 Related Work . . . 64
B.3 Description of our Approach . . . 66
B.3.1 Data Structure Description . . . 66
B.3.2 Rendering the Model . . . 67
B.3.3 Geological Concepts Through Sketches . . . 69
B.4 Results . . . 72
B.5 User Evaluation . . . 74
B.6 Conclusion . . . 77

C Rapid Modelling of Interactive Geological Illustrations with Faults and Compaction 79
C.1 Introduction . . . 81
C.2 Related Work . . . 82
C.3 Outline . . . 84
C.4 Modelling Approach . . . 84
C.4.1 Internal Representation . . . 85
C.4.2 Geological Features . . . 86
C.4.3 Alternative Visualizations . . . 91
C.4.4 GPU Realization . . . 92
C.5 Results . . . 92
C.6 Conclusions . . . 94

Part I

Overview

Chapter 1

Introduction

Geological models are generally abstract simplifications of natural systems, that aim to represent essential processes and properties. They provide a controlled environment in which geologists can work on, especially for interpretations or simulations.

Before the digital era, subsurface *geomorphology*, the study of geological shapes and the processes that originated them, was studied either in the field or in laboratories. Unfortunately, not every question can be answered with such approaches. One of the reasons is that the temporal scale of observation is completely different from the temporal scale of many geological processes. Direct observation is limited to years or decades, while geology, such as a stratigraphic rock layer, evolves over millions of years. For "short-term" geological events such as meandering rivers (in the order of centuries), this limitation can be partially overcome through the use of archived historical observations. Modelling can help geologists to address some of these issues and provide a complementary way of gaining insights into geological processes. Controllable and repeatable analyses can be conducted on models. Geomorphological modelling is a growing discipline. Many models have been developed for academic and commercial purposes, which are often quite different in their aims, assumptions and capabilities.

An important branch of geomorphology is structural geology. Structural geology is the study of the three-dimensional distribution of rock units (*lithostratigraphy*) with respect to their deformational histories. In this dissertation, we define how to create digital 3D models that express structural geology, as well as depositional and erosional processes in an illustrative manner. Several geological modelling operators are introduced to achieve our aim which is a quick and intuitive approach to geomodelling. Each geological operator is performed by sketching the essential shape that identifies the geological process or the subsurface configuration to be illustrated. That is, fast creation of illustrations is achieved by adopting sketch-based techniques tailored for expressing various geological configurations.

1.1 Problem Statement

Imagine we are in an important meeting. There is a group of participants that have to make a decision with respect to, for instance, the existence of natural resources. The existence of natural resources at an unexplored area is dependent on its geological configuration. An example is the Barents Sea and its subsurface lithostratigraphy, which is believed to contain hydrocarbons.

The oceanic crust of the Barents Sea presents some complexities in its sedimentation

structure. In this meeting, geologists gathered to discuss possible interpretations of the field area. The geologists have collected any kind of data at their disposal about the area and use any relevant information to converge into a global picture. An example of data at disposal in a qualitative manner is given by analogues from Svalbard [67, 53]. The island of Svalbard can be considered as being a part of the sea ground that has risen to above sea level due to tectonic forces, exposing an example of subsurface layering at one sample point at the Barents sea. Much research on the revealed geological structures at Svalbard has been performed prior to and independently of this meeting. Geologists can use this qualitative information to create hypotheses about the structure elsewhere in the Barents sea. There is an interpretation and discussion process, where it is important for everyone to be able to interact with the model to explore different alternative scenarios and their consequences. Subsurface geology consists of several shapes and, in most of the cases, it is not easy to say what kind of process has led to the current configuration. For that reason, a team of geologists is employed to share different expertises and opinions. Some of them are experts on folding processes, others in faults or fluvial systems. A common discussion on the same scenario brings a broader knowledge and increases the chances to develop the right interpretation of the involved processes.

In this first phase, it is important to externalize ideas and show them to the others. Moreover, it is also desirable that everyone can interact with what has been externalized by one person. In this way, such as when using a blackboard, weak points or mistakes can more easily be detected (by the presenter herself or by the listeners).

A geologist, with a specific expertise in the situation being examined, wants to express a point of view regarding the interpretation of a plausible process that generated the current configuration. Based on her experience on the subsurface, she wants to figure out ancient configurations of the area and the processes that were involved. And most important, the geologist needs to convince the others by explanations that have to be visually supported. Her initial thought is followed by comments and suggestions which most likely lead to an improved model. She may not have noticed the role of an ancient river channel which deposited in the Barents Sea area and contributed to accumulate sandstone and natural energy resources. Therefore she leaves the word to an expert in these fluvial system evolutions, who adds to the initial model all the channels and related deltas that may have contributed to the geomorphology development. Once all the involved geological features are placed in the model and a common view of the geomorphological history in the Barents Sea has been achieved, the geologists have another aim: they need to convince the management board that they have the right model on which to invest.

In this following meeting, the approach is quite different from the previous meeting, where peer experts were discussing together. Now one of the geologists speaks on behalf of all of them, bringing their common model to the board. This form of communication is one-directional, the leading geologist presents their ideas in a simple form, such that non-experts understand the essential concepts behind the natural processes and their consequences. The management will make a decision based on the opinions of the geologists, which have been condensed into illustrative models. The leading geologist cannot show field measurements to non-experts, because they would not understand, and cannot give the measurements to an algorithm that automatically produces a communicative animation of the geological event, because qualitative decisions must be taken when creating the result; neither can she show a set of existing template illustrations that in some way could be related to the actual real situation, because they would be to general. She

does not create the illustration with a professional 3D modelling software, as this takes too much time and requires detailed expertise in the modelling software. Moreover, for time reasons, she cannot generate an image that recreates the 3D perspective, illumination, textures and complex shapes with a 2D drawing software or with her hand-drawing abilities (which might not be too well developed). A physical representation of the event would be another option, but it is not realizable in practice. A feasible option would be to take a piece of paper and quickly draw to approximatively reproduce an abstract illustration of the model she has in mind. However, a digital representation would help the quality and the communicative power of the illustration. In addition, a digital representation is reproducible and editable, and the illustration that the geologists created as a common model can be proposed again to the board, possibly with minor modifications.

1.2 Requirement Analysis

In our hypothetical meetings, the person who explains processes concerning the Barents Sea would benefit from a rapid modelling technology, for prompt interaction, and a 3D illustrative visualization, to produce a simple abstraction which maintains the relevant information.

In general, scientific illustrations should fulfil different requirements, depending on the target audience:

1. When communicating towards peer experts, illustrations should be:
    - **technically correct**: small approximations are tolerated, but geological constraints cannot be discarded;
    - **quickly modelled**: when different points of view are needed, abstract ideas must be interactively shared;
    - **reproducible**: such that research can be spread in different venues.
2. For the scope of management decisions, illustrations should:
    - **be perfectly refined**: no space for weaknesses in the model, critical decisions are based on it;
    - **sell message**: benefits shall show through visual interpretation;
    - **impress viewer**: feelings are often the basis for a decision.
3. Illustrations that aim at students for teaching purposes should:
    - **be educational**: conceptual information enables a generalization of the model that can be used for similar cases, while peculiarities of the studied case extends student experience;
    - **include animation of processes**: time dominates geology and if it is reflected in the illustration, processes are more easily understood;
    - **aid in remembering material**: a visual picture of an event is more powerful for our mind than written words.

In category (1) we classify all the situations where the common scope is to produce a representation (or several alternatives) of a studied case that is a joint work of peer experts. The preliminary meeting carried out by the geologists discussing the Barents Sea geomorphology is an example of category (1).

The following meeting, where one geologist presents a common interpretation to the management board, falls into category (2).

We have not given an example of the third category in the previous section, however illustrations for teaching purposes are broadly used by teachers in their courses or by geologists in educational seminars. A typical usage of type (3) is for displaying definitions of geological processes or for modelling of real cases that can be explored in classes when going to the field is not an option, or for illustrating what has been observed in the field. It can be argued that animations do not only fit into this category.

An illustration, can have parts based on measured points or interpreted data. In such cases, the modelling is constrained. The focus in this thesis is on unconstrained conceptual sketches which may be influenced by qualitative knowledge of the area (for instance from analogues). Approaches for creating models, when sparse or dense measurements in the area exist, are described in the next related work chapter.

That is, there is a balance between detailed realistic models and simple-to-read abstract illustrations. As, for example, between photo-realistic and cartoon-like illustrations. By realistic conceptual illustrations, we refer to illustrations which are enhanced by shadows or rock-like textures, while abstract conceptual illustrations are enriched with symbolic textures and other non-realistic symbols, which geologists know and use. Realistic illustrations are good for communicating to those who do not understand the geological symbols and textures. On the other hand, symbolic illustrations can more clearly communicate different aspects and properties to geologists. The results of this thesis are more oriented towards geologist usage and non-photorealistic visualization, however we provide some examples of realistic visualizations in Paper A, where we have used non-symbolic textures which are photos of rocks that have been made tileable. In geological illustrations, the geometric aspect is central, but other characteristics help to improve the expressibility of a model or to achieve different results in order to highlight different properties. In this perspective, illustrations may be enhanced with colours, textures, text or symbols. The geometric aspect is tackled in the first part of the modelling process through procedural or sketch-based techniques, as defined in the next chapter. The visualization aspect is introduced in the latest stage of the illustrative modelling and consists of the utilization of colours, textures, text or symbols to mainly convey non-geometric properties.

Summarizing the requirements of all scopes (1), (2) and (3), we can conclude that a rapid modelling technology is desired to gain interaction in the communicative process.

In the next chapter, we introduce the reader to related work on modelling terrains and geology in the subsurface. Few of the aforementioned requirements are satisfied by existing techniques. This thesis attempts to satisfy most of the remaining ones.

Chapter 2

Modelling and Visualization in Geology - State of the Art

Illustrating by means of pen and paper is still a common practice for geologists who want to externalize their ideas. The digitalization of this process has not evolved enough to fulfil the requirements raised in the previous section. One currently employed approach is to draw 2D pictures with drawing software, forfeiting all the benefits of a 3D representation, such as interaction, rotation, depth information and cross-sectional views. On the other hand, there are rapid model generation techniques, described in the next sections, where modelling is not performed with an intuitive sketch-based modelling procedure, but rather through procedural modelling.

Procedural modelling covers some of the requirements of the previous section, however it cannot be employed for interactive discussions requiring quick modifications of the model. Procedural modelling is mainly currently used for creating the top surface of a geological model, i.e. the terrain, as described in this chapter. Moreover, procedural modelling is hard to control due to limited and often non-intuitive parameters when producing real case scenarios.

Sketch-based modelling, on the contrary, has quickly grown in the last ten years to support intuitive design in several fields, including geology. In general and not specifically for digital applications or for geology, drawing sketches helps:

- the process of developing new ideas;
- to explain ideas to others.

A sketch-based approach is useful to create models from scratch or to show an interpretation based on sparse data when this is available.

This chapter gives a broader overview on the state of the art concerning geometric modelling in geology, going through both procedural and sketch-based approaches.

Several modelling methods have been introduced to build terrains and subsurface geology. We classify previous work into a data oriented taxonomy, consisting of approaches not building on measured data, which we call *data-free*, approaches making use of sparsely sampled data, which we call *sparse-data* scenarios and approaches making use of densely sampled data such as volumetric measurements, which we call *dense-data* scenarios.

2.1 Geomodelling in Computer Graphics and in Geosciences

Realistic appearance of natural sceneries has been a key topic in computer graphics for many years. The outcome of this research primarily targets the film and gaming industries. The modelling is often procedural and can be constructed with little user intervention. In most cases, only the top surface is the final product of the modelling process, even if subsurface features have been taken into account during the modelling.

Parallel to this development, the modelling of geological structures has been developed from the geological domain. This modelling process usually requires heavy user involvement and substantial domain knowledge. The model creation can often take up to one year of intensive work. The modelling process also includes data acquisition from the site which is to be modelled. The result is usually a complex 3D model, consisting of a number of different subsurface structures.

The needs of the entertainment industry and the geoscientific domain are substantially different, although they represent similar natural phenomena. While the former one puts emphasis on interactive realistic visual appearance, the latter one focuses on structural realism, i.e. the correctness from the geoscientific point of view, while discarding rapidity. In recent years, research in geosciences have identified the importance of rapid prospects, i.e., fast geoscientific interpretations at early stages in exploration. For such a use, the extensive development period of a typical geological model becomes a severe limitation. Rapid prospects raise a need for rapid modelling approaches that are common practice in computer graphics terrain modelling.

While the terrain synthesis for entertaining industries is a product of content creation carried out by artists, the geoscientific models are created based on qualitative expert knowledge or actual measurements, and are done by geologists and other geoscientists based on a substantial level of background knowledge and experience. Moreover, when modelling based on measurements, the input data are either densely covering a certain spatial area, for instance by means of large-scale acoustic surveys, or consist of sparse input data that are completed by extrapolating known values over the areas where no measurements are taken. There is a multitude of approaches to sample geology, for instance from seismic surveys (2D cross-sections, 3D volumes or 4D time varying volumes), boreholes (1D curves) or virtual outcrops from LIDAR scans (3D textured surfaces) [112].

2.2 Geological Objects

The study of structural geology divides the subsurface into geo-bodies [58] of different categories. Central objects are layers, horizons, faults, folds, channels, deltas, salt domes and igneous intrusions.

Much of the research in geomodelling explores how to represent geological feature such as horizons, folds, faults and deposition. Deposition occurs when eroded particles of rock are brought by wind, water or gravity to a different place, where they accumulate to form a new rock layer. The subsurface is composed of a set of layers with distinct material composition. The surface which delimits two adjacent layers is known as a *horizon*. Deposition does not modify the structure of the existing horizons, but gives origin to a new rock layer instead. Two fundamental geological phenomena involve modification of the original structure of horizons: the process of folding and the process

of faulting. A fold is obtained when elastic layers of rock are compressed. It is defined as a permanent deformation of an originally flat layer that has been bent by forces acting in the crust of the Earth. Faults originate when forces that act on a specific layer are so strong that they overcome the rock's elasticity and yield a fracture. A horizon is therefore displaced and becomes discontinuous across a fault. Channels are remains of buried river depositions.

2.3 Modelling of Geological Objects

Geological models can be divided into two different categories, *layer-based models* and *complex models* [133]. The layer-based models, built of multiple horizontal oriented surfaces, are typically created to model sedimentary geological environments for the purpose of ground-water mapping, or oil and gas exploration. In regions with complex geological structures or where the layering is not dominant, for instance when modelling igneous and metamorphic terrains, a more complex terrain model is needed. These complex terrains are modelled when exploring for metal and mineral resources. In this dissertation, we focus on the layer-based models.

In two viewpoint articles, Turner [133, 132] provides a thorough introduction to the challenges of creating computer tools for modelling and visualization of geological models. The articles formulate the essential domain needs and the capability to interactively model and visualize: geometry of rock and time-stratigraphic units; spatial and temporal relationships between geo-objects; variation in internal composition of geo-objects; displacement and distortions by tectonic forces; and the fluid flow through rock units.

Furthermore the following characteristics of the geo-bodies are highlighted: complex geometry and topology, scale dependency and hierarchical relationships, indistinct boundaries defined by complex spatial variations, and the intrinsic heterogeneity and anisotropy of most subsurface features. These characteristics are, according to Turner, not possible to satisfy with traditional CAD-based modelling tools. Thus, dedicated geological modelling and visualization tools are necessary. Marroquim et al. [80], for instance, propose a representation that is particularly suitable for geological models, where they use tetrahedra to describe the subsurface geology.

In geological modelling there are often scenarios that lack sufficient data. In order to build a meaningful model, the creator must interpolate between the sparse sampled or derived data available, or can sketch from imagination in case of no data. Traditional interpolation schemes for discrete signal reconstruction are not sufficient as the process needs to be guided by geological knowledge, often through many iterations, to produce a successful result. A plausible geological scenario has to follow certain geo-physical constraints. Caumon et al. [23] describe specific structural modelling rules for geological surfaces defining boundaries between different lithological layers. Geo-bodies exhibit spatial continuity, therefore abrupt geometric variations such as sudden change of normal orientation on the surface, and abrupt changes within a fault are not common. This implies that a structural model may be validated via reconstructing its depositional state.

In Section 2.4 we will briefly categorize previous works in terms of the type of data that is being addressed (see Figure 2.1).

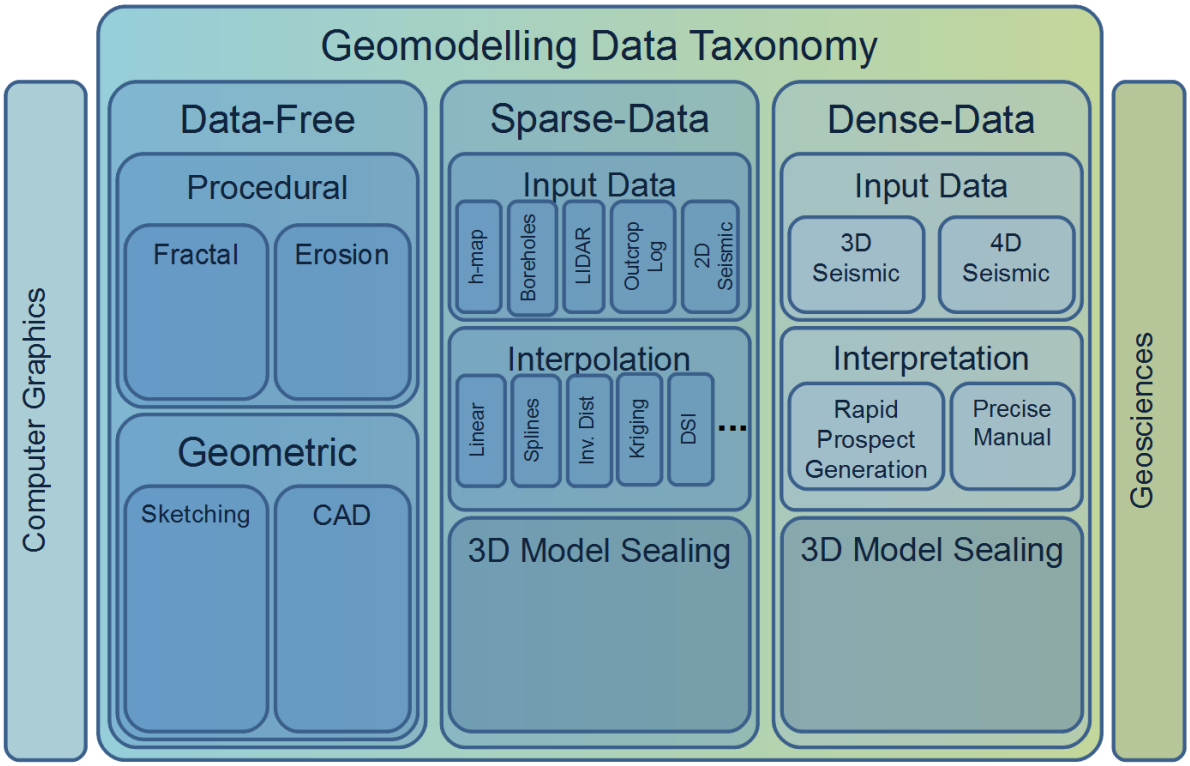


Figure 2.1: Geomodelling methods are classified according to which type of data they use and from which domain they originate (computer graphics to the left or geosciences to the right).

2.4 Geomodelling Data Taxonomy

We have decided to divide the literature of geologic modelling into three categories (see columns in Figure 2.1) according to how much measured data the model is based on. The *data-free* scenario represents current and future trends in rapid modelling, while the other two categories (*sparse-data* and *dense-data* scenario) are mostly faced with modelling that is constrained by acquired data. In the first category (no data) very little work has been done in geosciences, whereas computer graphics has contributed here due to the purely aesthetical need for creating realistic looking models for use in animations and games. Also general methods for sketching and modelling has been developed in the field of computer graphics. For sparse data, there are contributions from both computer graphics and geosciences. For instance, for interpolating scattered points there are Spline methods from computer graphics and mathematics, whereas the Kriging interpolation originated in the geosciences. For dense data, most of the research has been performed in the geosciences.

In the workflow where no measured data is required as input (data-free), some papers focus on surface creation, other general papers describe different mathematical surface representations. Several sketch-based papers describe different ways of fast sketching and assembling of solid objects and some focus on compact representations and fast rendering of complex solid objects. For the case where data is being used (sparse or dense-data), the workflow follows the columns in Figure 2.1 from top to bottom, beginning with measuring data, interpreting relevant structures, interpolating these into higher-order objects and representing these in some appropriate mathematical way. The structures

are then assembled into solid geometry describing the subsurface.

2.4.1 Data-Free Scenario

This section describes works that do not rely on any measured or sampled input data, and where the models are created from scratch, driven by imagination or concept ideas and domain knowledge.

The *data-free* scenario has no ground truth information and therefore the geometric synthesis relies entirely on *procedural* and *geometric* modelling. The typical computer graphics research agenda proposes methodologies that alleviate the user from labour-intensive tasks by automating parts of the modelling. Procedural modelling offers the modeller specific, easy to handle, input parameters which control the process of geometry generation. The geometry typically represents terrain surfaces. Procedural techniques in modelling has been facilitated mainly through *fractal* modelling [121, 9]. Simple *erosion* models [68, 57] are utilized to create dynamic and realistic landscapes [113, 59].

The shortcoming of procedural modelling is usually the lack of direct control over the landscape development. The modeller has a rough idea of the landscape, but implicit parameter settings do not guarantee a match with a modeller's idea of an intended shape. Therefore a combination of explicit geometric modelling, to represent the modeller's expectations, with procedural modelling, to add realism, is a preferred strategy. On the other hand, geometric modelling can be a labour-intensive task. For rapid modelling scenarios, various forms of *sketching* metaphors [48, 105, 138] or modelling by example [18] provide fast ways to express the rough structure of a terrain or, more general, of a stratigraphic model.

Procedural Modelling: Fractal and Erosion Surface Creation

Currently there seems to be three approaches to generate synthetic terrains: fractal landscape modelling, physical erosion simulation and terrain synthesis from images or sample terrain patches. Before the work by Olsen [93] (2004), it was mostly common to use simple fractal noise to obtain terrain surfaces, as computers were not fast enough to simulate erosion processes in real-time. Olsen proposes a synthesized fractal terrain and applies an erosion algorithm to this. His representation of terrains is a two-dimensional height-map. To simulate erosion, he considers the terrain slope as one of the main parameters: a high slope results in more erosion, a low value produces less erosion. Starting from a noisy surface, called the *base surface*, erosion occurs to simulate weathering on a terrain.

The ability to model and render piles of rocks without repetitive patterns is one of the achievements of the work by Peytavie et al. [106]. They focus on rocks and stones, which are found everywhere in landscapes. They provide realism to the scene, reveal characteristics of the environment and hint on its age. Before this paper, the standard way of generating rocks was to produce a few models by artists, which were then instantiated in the scene. To create piles of rocks, collision detection techniques were applied with a high computational cost and low control.

Musgrave et al. [87] describe the creation of fractal terrain models, avoiding global smoothness and symmetry; these two drawbacks arise from the employment of the first definition of fractional Brownian motion (fBm) as introduced by Mandelbrot [79]. Moreover, in their method there is a second stage in which the surface undergoes an approx-

imation of a physical erosion process. Terrain patches are represented as height-maps and the erosion process is subdivided into a thermal and a hydraulic part.

Concerning modelling terrains with rivers, Sapozhnikov et al. state that at the time when their paper [120] was written (1993), it was impossible to simulate the process of natural river network formation without making a substantial approximation; i.e. a simpler model that does not make use of physical laws, but nevertheless reproduces the main geometrical features of a real river network. They use a random walk method to generate a set of river networks of various sizes.

Stachniak et al. [126] point out that fractal methods have been used to create terrain models, but these techniques lack control by the user. They try to overcome this by imposing constraints to the original randomly created model, according to the user's wishes. The method requires two inputs: the initial fractal approximation of the terrain and a function that incorporates the constraints to be satisfied in order to achieve the final shape. As an example, they show how to adapt a fractal terrain to accommodate an S-shaped flat region, representing a road. The constraint function defines a measure that indicates how close a terrain is to the desired shape. The final solution is provided by a minimization of the difference from the current terrain to the desired one.

Another way of combining procedural modelling with user constraints is described by Doran and Parberry [40]. In their work, they procedurally generate terrain elevation height-maps, taking into consideration input properties defined by the user. The model lets the user choose amongst five terrain tools: coastline, smoothing, beach, mountain and a river tool. Together, they can generate various types of landscapes.

A terrain surface is created by fractal noise synthesis in Schneider et al.'s work [121]. They aim to solve the problem that was one of the biggest disadvantages in fractal terrain generation at the time (2006), namely the setting of parameters. They reduce such an unintuitive process of setting parameters by presenting an interactive fractal landscape synthesizer.

Roudier et al. [119] propose a method for terrain evolution in landscape synthesis. Starting from an initial topographic surface, given by a height-map, they subsequently apply an erosion process to obtain the final 3D model. The erosion consists of mechanical erosion, chemical dissolution and alluvial deposition.

Chiba et al. [27] propose a method that overcomes the limitation of previous techniques for generating realistic terrains through fractal-based algorithms. However the method lacks ease of handling, i.e. it is not possible to modify the surface on the basis of the user's suggestions such as introducing ridge or valley lines, which are usually important to characterize a mountain scenery. The topology of the landscape is created by a quasi-physically based method, that produces erosion by taking velocity fields of water flow into consideration. The whole process of erosion, transportation and deposition is derived on the basis of the velocity field. On the other hand, Dorsey et al. [41] focus on the visual effect of erosion on a single stone or rock, represented volumetrically, taking into consideration weathering effects on it.

In the work by Benes et al. [10], a method for eroding terrains is described. A concise version of a voxel representation is utilized, and thermal weathering is simulated to erode the initial model. This new way to represent terrains has the advantage of being able to represent caves and holes. When applying erosion, all the layers, including the ceilings of the caves, are involved in the process.

In a subsequent paper [12], a technique for procedural modelling of terrains through hydraulic erosion is introduced. The purpose is to use a physical-based approach together

with a high level of control. Contrary to previous techniques which tend to oscillate during water transportation, they provide a tool for hydraulic erosion that is fast and stable. They overcome oscillation, relying more on physical constraints than was previously the case. The erosion process consists of four independent steps, where each step can run repeatedly and in any order. These four steps are: introduction of new water (simulation of rain); material capture by water (erosion); transportation of material; and deposition at a different location.

Another work by Benes et al. [13] applies a hydraulic erosion fully based on fluid mechanics using the Navier–Stokes equations that describe the dynamics of their studied models. They use a 3D representation provided by a voxel grid and the erosion process leads to a model that can either show a static scene or an animation illustrating the evolution of terrain morphology. At each iteration of the erosion process, a solution to the Navier–Stokes equations is computed to determine a pressure and velocity field in each voxel.

Interactive physics-based erosion is employed by Stava et al. [127] (Figure 2.2). This work is based on hydraulic erosion, and achieves interactivity, which allows the user to take an active part during the generation of the terrain. The technique is implemented on the GPU, and because of limited GPU memory, the terrain is subdivided into tiles, which individually fit in GPU memory. Each terrain is represented as a height-map.

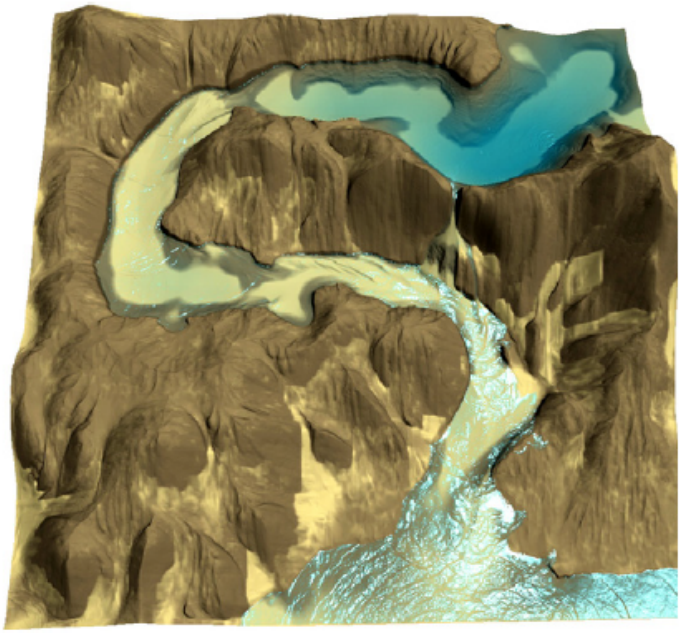


Figure 2.2: The eroded terrain is obtained by simulating the movement of the water flow and transportation of rock particles [127].

Kristof et al. [68] adopt 3D terrain modelling through hydraulic erosion by fluid simulation using a Lagrangian approach. Smoothed Particle Hydrodynamics (SPH) [51, 75] is employed in this paper to solve dynamics that generate erosion. SPH require low memory consumption, acts locally, works for 3D features and is fast enough to work on large terrains.

For Hnaidi et al. [57], the terrain is generated from some initial parametrized curves which express features of the target terrain (see Figure 2.3). Constraints such as elevation and slope angle are assigned to each curve.

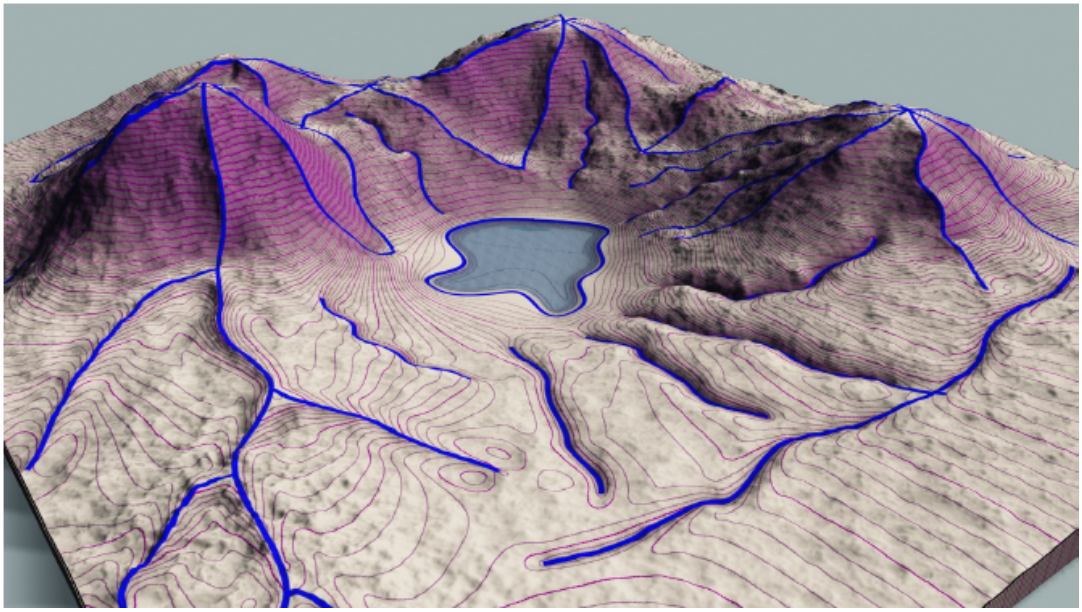


Figure 2.3: Sketches visible in the figure as blue strokes work as constrains during the method proposed by Hnaidi et al. [57].

Prusinkiewicz and Hammel [113] address the problem of generating fractal mountain landscapes, which also includes rivers. They do this by combining a midpoint-displacement method for the generation of mountains with a method to define river paths.

Hudak and Durikovic [59] tackle the problem of simulating terrain erosion over a long time period. They use a particle system and take into consideration that terrain particles can contain water. Discrete Element Method (DEM) is used for the simulation of the soil material and Smoothed Particle Hydrodynamics (SPH) simulates water particles.

Instead of procedural and erosion synthesis, Brosz et al.'s paper [18] introduces a way to create realistic terrains from reference examples. This process is faster than starting the terrain generation from scratch. Two types of terrains are necessary to obtain the final one: a base terrain, used as a rough estimate of the result, and the target terrain that contains small-scale characteristics that the user wants to include in the reconstruction.

Brosz et al. present two ways to generate landscapes represented as a height-map: using brush operations to bring some predefined information or action on the surface; alternatively, simulation and procedural synthesis can be applied to obtain a realistic terrain. One drawback of using simulation is that it can be slow, while in the case of procedural synthesis, expressibility is reduced by a limited set of parameters. De Carpentier and Bidarra try to combine brushing and procedural synthesis in their work [36], an example is shown in Figure 2.4.

Geometric Modelling

Gain et al. present a paper [48] that describes procedural terrain generation using a sketching interface. Their approach aims to gather benefits, such as intuitiveness, and overcome some limitations, such as height definition, of previous methods of sketch-based terrain modelling [29, 141, 147]. Watanabe and Igarashi [141] use straight lines and, even though they yield a boundary for landforms using local minima and maxima of the user's sketch, they do not give the user the possibility to change the proposed

Figure 2.4: Two types of noise as seen on left and right side applied by de Carpentier and Bidarra [36] to achieve a realistic terrain.

shape. Furthermore, they apply noise onto the terrain after surface deformation, hence the obtained surface does not interpolate the user strokes exactly. Whereas landforms rarely follow straight lines, Zhou et al. [147] allow landforms to have more freely shaped paths using a height-map sketching technique as guidance for an example-based texture synthesis of terrain. In contrast to the method suggested by Gain et al. [48], they provide low and indirect control over the height and boundary of the resulting landforms.

Brazil et al. [138] introduce a sketch-based technique to generate general 3D closed objects using implicit functions. They also show how to obtain simple geological landscapes from few user strokes, see Figure 2.5. Several of the fractal and erosional surface creation methods represent the surfaces as height-maps. This is an easy-to-maintain data structure which fits well with erosional calculations. However, the method by Brazil et al. [138] can represent complex surfaces with overhangs or closed objects, using implicit functions defined as a sum of radial basis functions. Based on points with normals as input, a smooth implicit function, interpolating the points while being orthogonal to the normals, is created.

Peytavie et al. [105] represent complex terrains with overhangs, arches and caves. They achieve this by combining a discrete volumetric representation, which stores different kinds of material, with an implicit representation for the modelling and reconstruction of the model.

Bernhardt et al. [14] presented a sketch-based modelling tool to build complex and high-resolution terrains. They achieved a real-time terrain modelling by using both CPU and GPU calculations. To represent large terrains, they use an adaptive quad-tree data structure which is tessellated on the GPU.

In the following text, we consider solid representations as being different from boundary representations in that they are not hollow, but have spatially varying properties. Takayama et al. [130] present diffusion surfaces as an extension of diffusion curves [97] to 3D volumes. The representation consists of a set of coloured surfaces in 3D. A smooth volumetric colour distribution that fills the model is obtained by diffusing colours from these surfaces. Colours are interpolated only locally at the user-defined cross-sections using a modified version of the positive mean value coordinates algorithm. A result of

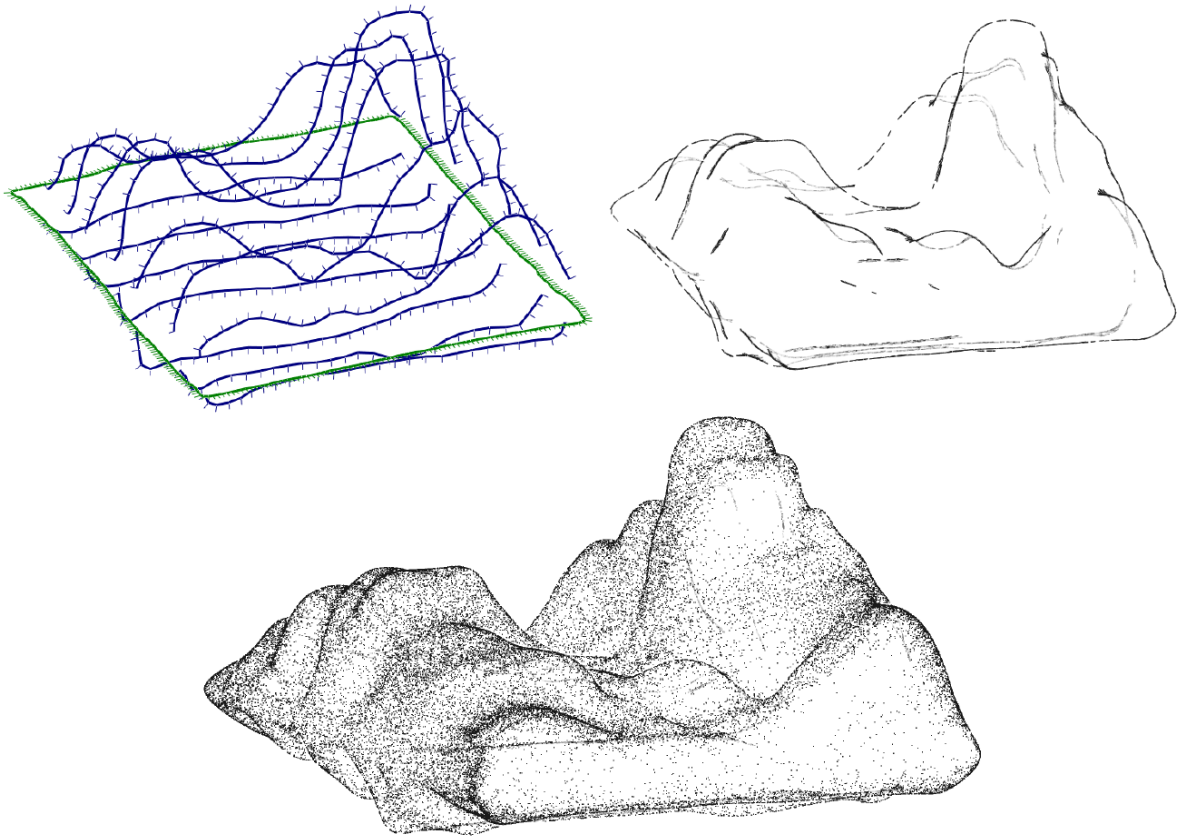


Figure 2.5: Landscape example generated with the sketch-based approach by Brazil et al. [138]. Top left image shows the input curves and their normals. Right and bottom images show resulting landscape in two different rendering styles. The model is represented with Hermite Radial Basis Functions.

the work by Takayama et al. [130] is shown in Figure 2.6.

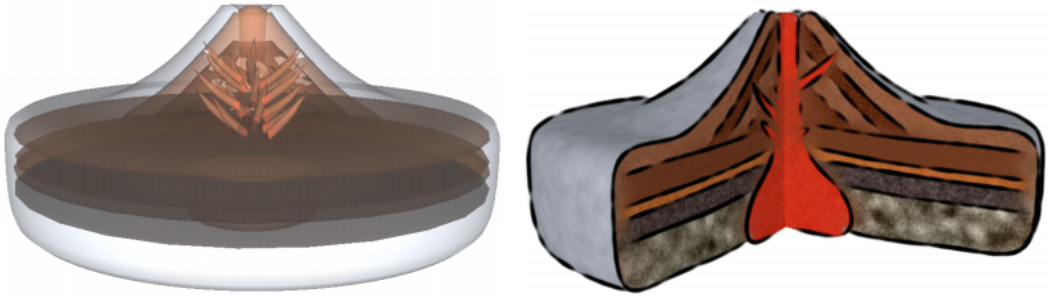


Figure 2.6: A volumetric representation of a geological scenario using diffusion surfaces (Takayama et al. [130]).

In the work by Wang et al. [140], objects are represented as implicit functions using signed distance functions. Composite objects are created by combining implicit functions in a tree structure. This makes it possible to produce volumes made of many smaller inner components. This multi-structure framework lets them produce models irrespective of resolution (see Figure 2.7 for a geological application example).

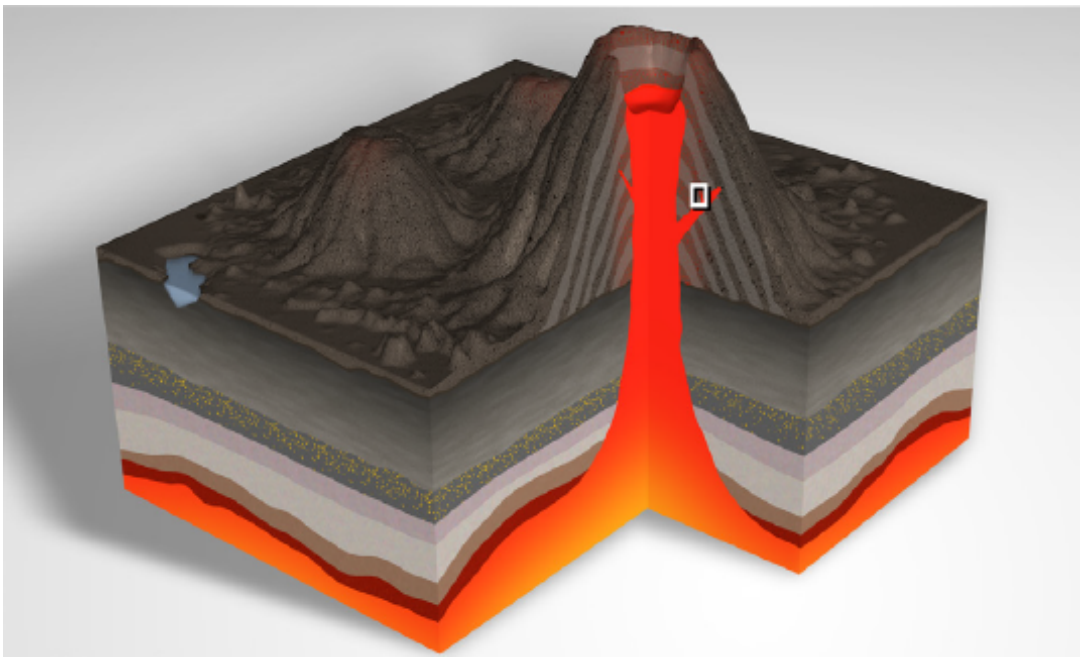


Figure 2.7: A volumetric representation of a geological scenario using an implicit representation (Wang et al. [140]).

2.4.2 Sparse and Dense Data Scenarios

This section describes methods that use sparsely scattered geologically measured data or dense data, such as 3D seismic reflection volumes, for creating a subsurface model. In contrast to data-free modelling, the process is now naturally constrained by values in the data.

The *sparse-data* modelling scenario is the most frequently used in the geoscience domain [52]. Very often data comes from boreholes [69, 65, 84], where the data needs to be interpolated between. Besides boreholes, there are often other acquisition types available, such as surface elevation models [39, 146], obtained through the process of remote sensing. This heterogeneous pool of geoscientific data raises the challenge of data integration and data interpolation. The main interpolating methods are the B-Spline method, the inverse distance method, the Kriging method and the discrete smooth interpolation method [78, 76]. We will briefly discuss them later in this Section. The implicit function interpolator is another increasingly popular interpolation method [82].

Turner [133] demonstrates how to build a typical geological model from a *sparse-data* scenario by firstly interpreting bore-hole logs to construct triangulated surfaces of horizons [8] and then create the geo-bodies [37, 125] in sealed, boundary-representations [24] of the volumes between these surfaces. The modelling of faults [143] is also very important and Turner describes the challenge of modelling the interface between the boundary representation and the fault to avoid unwanted crossings or empty spaces between faults and geo-bodies. Using a structured mesh representation of the boundary surface can result in discretization errors, while using unstructured grid representation [47] adds computational complexity and results in slower model construction.

The *dense-data* scenario is typically based on a single- or multi-attribute seismic dataset. The first challenge is purely of computational character, i.e., how to interactively display large amounts of volumetric data. This was addressed, e.g., in the work

of Plate et al. [109]. Utilizing volume rendering, such datasets can be displayed without prior extraction of geo-bodies. Extracting geological structures from such a dataset is necessary for consecutive steps along the workflow, such as reservoir modelling. This process is known as geoscientific interpretation and is a very time-demanding task. Typically the original seismic dataset, consisting of the amplitudes of reflected sound waves, can be used to extract a number of derived attributes. These attributes are not geo-bodies, but their distribution over the 3D domain indicates the presence of certain geological structures. The *SHIVR* system [72] can extract geo-bodies based on scatter plots, as shown by Andersen and van Wijngaarden [6]. Rapid prospect generation can certainly benefit from faster interpretation. Patel et al. proposed methods for rapid horizon extraction in two [102] and three dimensions [101]. Afterwards, once the interpretation is available, 3D visualization can assist in validating the correctness of the extracted horizons with respect to original or derived attribute data [103].

A natural next step after 3D structural modelling is the development of a time-varying structural model. Inverse methods are often utilized in geomodelling to restore hypothetical geological scenario backwards in time [142, 54, 22]. Such an approach aims at restoration of deposited sedimentary layers, for example through unfolding. Restored information about palaeogeography often gives good indication where to search for hydrocarbon reservoirs.

Measured Data

Subsurface data can be collected in several ways, at various effort and expense. Seismic 2D or 3D reflection data is collected by sending sound waves into the ground and analysing the echoes. When the sound waves enter a new material with a different impedance, a fraction of the energy is reflected. Therefore, various layer boundaries of different reflective character are visible in the seismic data as linear edges. Well logs are obtained by drilling into the ground while performing measurements and collecting material samples from the well. Outcrops are recorded by laser scans together with photography (LIDAR) to create a 3D pointcloud of the side surface of geology [112]. This surface can be investigated and visible layer boundaries can be identified and outlined as curves along the surface.

Interpolation

Key interpolating methods for surfaces in geosciences are the B-Spline method, the inverse distance method, the Kriging method, the Discrete Smooth Interpolation (DSI) method [77, 78, 76] and the Natural Neighbor Interpolation method [122]. Kriging is a statistical approach to interpolation that incorporates domain knowledge and is uncertainty-explicit [28, 134]. Kriging, like exemplar-based synthesis, creates a surface that has similar properties to an example dataset.

The Discrete Smooth Interpolation allows for integration of geo-physical constraints into the interpolation process. The interpolator takes as input a set of (x, y) positions, some with height values and others without. After interpolation, all positions have been given height values. Discontinuities between positions can be defined so that certain neighbour points do not interpolate. Typically for a horizon surface, discontinuities would be added across faults. In addition, constraints such as having points being attracted towards other points, having points being limited to movement along predefined lines or on surfaces can also be defined. These constraints are useful for interpolating

geologic surface data. However the method might not be well suited for cases with very little or no observation data (as indicated by De Kemp and Sprague [37]), such as in the data-free scenario. Natural Neighbor Interpolation is also based on a weighted average, but only of the immediate neighbours around the position to be interpolated. A Voronoi partition is created around all known points and the weight is related to the area of these partitions around the unknown point.

An interpolation and surface representation system for geology is discussed in the work by Floater et al. [45]. Scattered point measurements can come in many forms, uniformly scattered, scattered in clusters, along measurement lines or along iso-curves. Fitting a surface through the points requires interpolation. Different interpolation methods vary in quality dependent on the distribution of the scatter data. Floater et al. offer interpolation in form of piecewise polynomials on triangulations, radial basis functions or least squares approximations.

Although more of a connectivity algorithm than an interpolation algorithm, the work of Ming and Pan [84] presents a method for constructing horizons from borehole data. Each borehole dataset consists of a sequence of regions. Each region has its start and end depth specified as well as its rock type. One rock type might appear in several layers and also the rock type sequence might vary between boreholes. This results in several possible connectivity solutions. The challenge is to make a suitable matching of layers to create a solid layer for each rock type.

Interpretation

Several commercial tools exist for interpreting 3D seismic data. One example is Petrel [123] where the user sets seed points and the system grows out a surface. The user can change the growing criteria or the seed points until a satisfactory surface is extracted. This can be time consuming. Kadlec et al. [62] present a system where the user interactively steers the growing parameters to guide the segmentation instead of waiting until the growing is finished before being able to investigate it. Fast extraction of horizon surfaces is the focus of Patel et al. [101]. Their paper introduces the concept of brute-force and therefore time-consuming preprocessing for extracting possible structure candidates in 3D seismic reflection volume. After preprocessing, however, the user can quickly construct horizon surfaces by selecting appropriate candidates from the preprocessed data. Compact storage of all surface candidates is achieved by using a single volumetric distance field representation that builds on the assumption that surfaces do not intersect each other. This representation also opens up for fast intersection testing for picking horizons and for high quality visualization of the surfaces. The system allows the user to choose among precomputed candidates, but editing existing surfaces is not possible. Editing is addressed by Parks [99] and Amorim et al. [5]. They present methods that allow to quickly modify a segmented geologic horizon and to cut it for modelling faults.

Free-form modelling is achieved using boundary constraint modelling [16]; this is simpler and more direct than Spline modelling, which requires manipulation of many control points. Discontinuities arising from faults are created by cutting the mesh. Amorim et al. [5] allow for more advanced surface manipulation in their system. Surfaces with adaptive resolution can be altered and cut with several sketch-based metaphors. In addition, the sketching takes into account the underlying 3D seismic so that it can automatically detect strong reflection signals which may indicate horizons.

3D Model Sealing

Solid geometric representations of subsurface structures is important for analysis. A sealed model enables consistent inside/outside tests, providing well defined regions. It is also the first step for producing physical reservoir simulations of liquid or gas flow inside the model at later stages.

Baojun et al. [8] suggest a workflow for creating a 3D geological model from borehole data using commercial tools and standards. They use ArcGIS for creating interpolated surfaces from the sparse data. They use geological relevant interpolation such as Inverse Distance Weighted, Natural Neighbor, or Kriging interpolation. This approach results in a collection of height-maps which are imported into 3D Studio Max and stacked into a layer cake model. Then Constructive Solid Geometry (CSG) [111] operators are used to create holes (by boolean subtraction) at places where data is missing in the well logs. The model is then saved as VRML enabling widespread dissemination through viewing the model in web browsers.

Lemon and Jones [69] present an approach for generating solid models from borehole data (see Figure 2.8). Boundary points in the borehole data is interpolated into surfaces. For creating a closed model, they state and exemplify that CSG together with set operations can be problematic as the set operation trees grow quickly with increased model complexity. They simplify the model construction by representing horizons as triangulated surfaces and letting all horizon vertices have the same set of (x, y) positions and only varying the z positions (see Figure 2.9). This simplifies intersection testing between horizons and makes it trivial to pairwise close horizons by triangulating around their outer borders.

Complexity increases when models must incorporate discontinuities in the layers due to the faults. Wu and Xu [143] describe the spatial interrelations between faults and horizons using a graph with horizons and faults as nodes. The graph is used to find relevant intersections and bounding surfaces which are Delaunay triangulated to form closed bodies. In a follow-up paper [43], two types of fault modelling techniques are compared (based on what they call *stratum recovery* and *interpolations in subareas*) and a unified modelling technique for layers and faults is presented to solve the problems of reverse faults (i.e. convergent sedimentation blocks), syn-sedimentary faults (when slumping of sedimentary material happens before it is lithified) and faults terminated inside the model (also known as blind faults).

Solid modelling tools in CAD do not easily support subsurface features such as hanging edges and surface patches. Many papers describe data structures for representing the solid blocks that horizons and faults subdivide the subsurface into. Boundary representations are frequently used. Generalized maps, used for describing closed geological models, are introduced by Halbwachs and Hjelle [55].

Implicit surfaces (implicits) provide a suitable way to represent geological solids [82]. Essentially, such solids are described by implicit functions that can be expressed in different forms, e.g., distance based models, analytical functions or interpolation schemes like for instance RBFs. Pasko et al. [100] generalized the above representations, which lead to an inequality, $f \geq 0$, also called *functional representation* of solids. Kartasheva et al. [64] introduced a robust framework to model complex heterogeneous solids, which was based on functional representation. The implicit solid definition is quite broad, and for instance, the terrain modelling using a height-map can easily be represented by implicits [49].

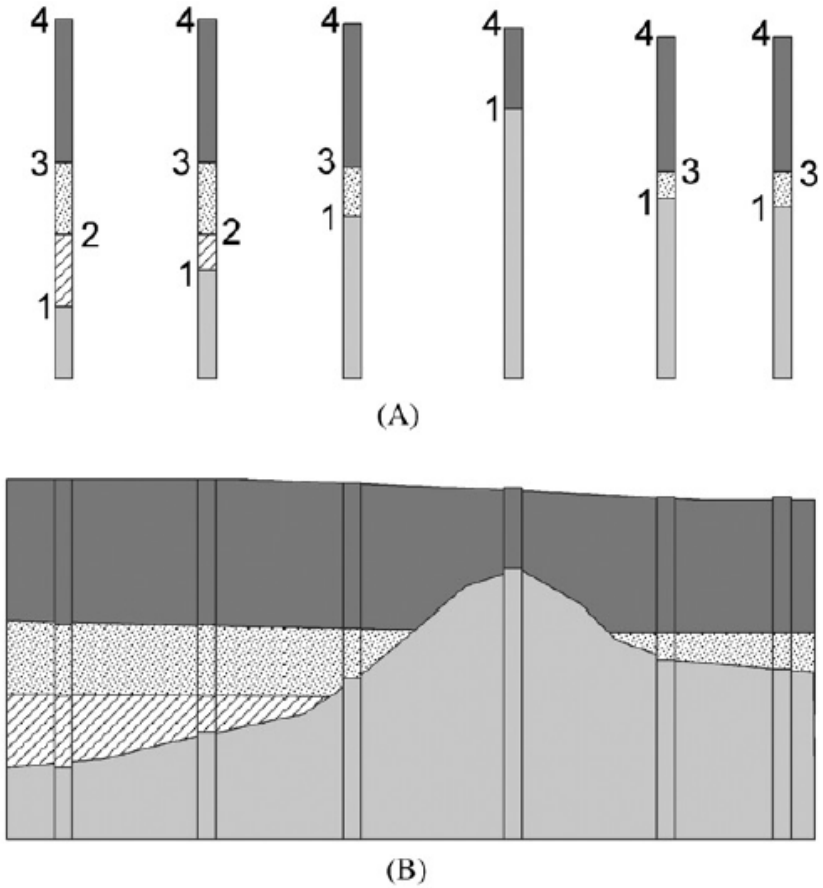


Figure 2.8: Borehole data with identified layer boundaries in a) and resulting interpolation in b) from the method by Lemon and Jones [69].

2.5 Illustrative Visualization

Illustrative 3D visualization has mostly been researched in fields such as medicine [137], but not in geology. For a wide overview on illustrative visualization techniques and their application to several fields, we suggest the tutorial by Viola and colleagues [137]. When considering medical illustrations, Sousa et al. [124] present a volume illustration method for interactive simulation sessions. While in geological illustrative visualization, Patel et al. [103] propose an approach to display volumetric seismic data. The level of abstraction provided by illustrative visualizations is used when communicating interpreted or simplified scenarios, e.g. when people with different background are present in the audience.

2.6 Challenges and trends in geological modelling

Geoscience technology on closed model representations and model updating has not progressed at the same speed as in computer graphics. Better knowledge transfer between computer graphics and geosciences would be advantageous. Caumon et al. [23] state that beginners with 3D modelling too often lose their critical sense about their work, mostly due to a combined effect of well defined graphics and non-optimal human-machine communication. It is also important that a structural model can be updated when new

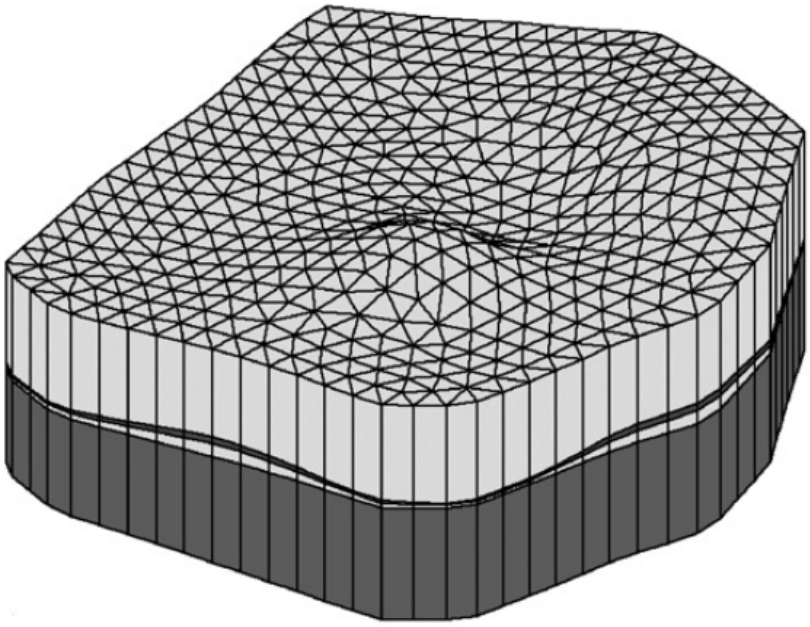


Figure 2.9: Example of model created with method by Lemon and Jones [69]. The shared (x,y) vertex positions can be seen on the side surfaces.

data becomes available or perturbed, to account for structural uncertainties. In other words, with current modelling technology, uncertainty is difficult to express, and models are hard to update.

Researched literature from geosciences emphasizes a strong need for modelling technology for communication and further analysis of the Earth's subsurface. While several matured methods are now in use by the domains of geology and geosciences, all tools require considerable effort to build structural model. Current tools focus on precise modelling in favour of rapid modelling. But rapid modelling is the key for the ability of expertise exchange, especially in the early phases of the interpretation process, which is the focus of the thesis.

In this direction, we contribute to cover missing technology, necessary to fulfil all the requirements highlighted by the meeting on the Barents Sea. In the next chapter, we describe how.

Chapter 3

Modelling and Visualization - Extending the State of the Art

Participants of the meeting are now discussing future strategies for an area of the Norwegian side of the Barents Sea that has not yet been digitized or measured. They need to create a complete geological model and illustration from scratch.

Here we give a description of the methods that we introduced in our work to rapid model and visualize illustrations. All three papers A, B and C are based on producing a layer-cake representation as a final model. The layer-cake model is a typical form for conveying subsurface structures. It is broadly employed by scientists in academia and industry.

The focus of what we model changes a bit through the papers to cover different aspects. In the first approach, Paper A, we create a 3D model by sketching on a cross-section and extruding the curves into surfaces. From the boundary of those surfaces, we seal the volumetric model with a triangulation. One of the advantages of having a mesh representation is that we can easily apply textures on the model. For warping the textures to fit the layer shape, we use conformal mapping, which preserves angles. This is important when deforming an illustrative texture that needs to be recognized (either because of common use amongst geologists or because it has been defined in an accompanying legend).

In the following work, reported in Paper B, we choose a different point of view for the user to sketch, namely from top-view. This is more appropriate when dealing with fluvial systems and their morphological evolution. To be able to encode more details in river and delta deposition processes, a novel concise representation of layers was introduced. Layers are still defined by a boundary representation in the modelling stage, i.e. a heightmap is assigned to each layer of the illustration. During rendering, we treat the model as a volume, obtained by our proposed ray-casting technique on the heightmap representation. That opens up for more versatile rendering and permits to internally inspect generated illustrations.

In our last work, Paper C, we still use the heightmap representation with volumetric rendering, but extend it, so that it can incorporate fault processes as well. We then show that the volumetric model is suited to be internally explored in several ways, depending on the main focus of the viewer. GPU acceleration is exploited to manage a higher computational cost due to the introduction of interactively animated faults.

3.1 Contributions

Our contribution leads to a new approach of designing 3D illustrations, which can be utilized during discussions between experts, for teaching purposes or for industrial strategic decisions. It is suitable for different scopes, but has a common achievement, that is abstraction and modelling (i.e. going from a real case to a virtual model) for simplification and interpretation.

We provide new visualization and modelling methods to digitally express an abstract concept related to subsurface morphologies. With the help of a sketch-based approach, simulating the intuitive way of expressing ideas through pen and paper, and by visualizing obtained models with illustrative techniques, we are able to generate 3D illustrations and animations in little time and less effort compared to alternative techniques.

In this dissertation, we contribute to extend the state of the art in modelling and visualization with the following aspects:

- Rapid Modelling (Section 3.2)
- Illustrative Animated Storytelling (Section 3.3)
- Layered Data Representation for Morphological Evolution (Section 3.4)
- Model Discontinuities and Interactive Illustrations (Section 3.5)

3.2 Rapid Modelling

An example of sketch based solid assembly of geological layer-cake models is presented in Paper A. Here we describe how to obtain a boundary representation of a solid model with the use of a sketch-based technique. Our approach lets a user sketch layers on the side of a bounding box which is extruded to a solid.

To outline our core contribution of Paper A, we show Figure 3.1, which lists three simple actions defining some of the most common processes in geology: folding, faulting and erosion. Folding and erosion events are distinguishable by a different texture deformation.

We compute 3D surfaces as extrusions in the third dimension of the drawn curves and represent them as triangular meshes. Adjacent surfaces are sealed together by triangulating their boundaries, hence creating closed layers. The shape of folded layers is thus given by the shape of the sketched curves. Whereas layer discontinuities are defined by a cross-sectional sketch and a further user input curve. From these curves, we deduce direction and amplitude of displacement. To change the model, the user can select a previously sketched curve and redefine parts or all of it by over-sketching.

The illustrative message of the model is enhanced by textures which are attached to the sides of the layers. Professional illustrators are trained to recognize standardized illustrative textures or non-standard textures together with a texture legend. In either case, the role of a texture is important to convey the types and properties of layers that are present in the model. We automatically adapt a texture to the shape of a layer, using a conformal map, that by definition preserves angles. This latter property is often important because we do not want to alter the illustrative meaning of the input texture. In addition, the texture on each layer can be reshaped independently of the others. The

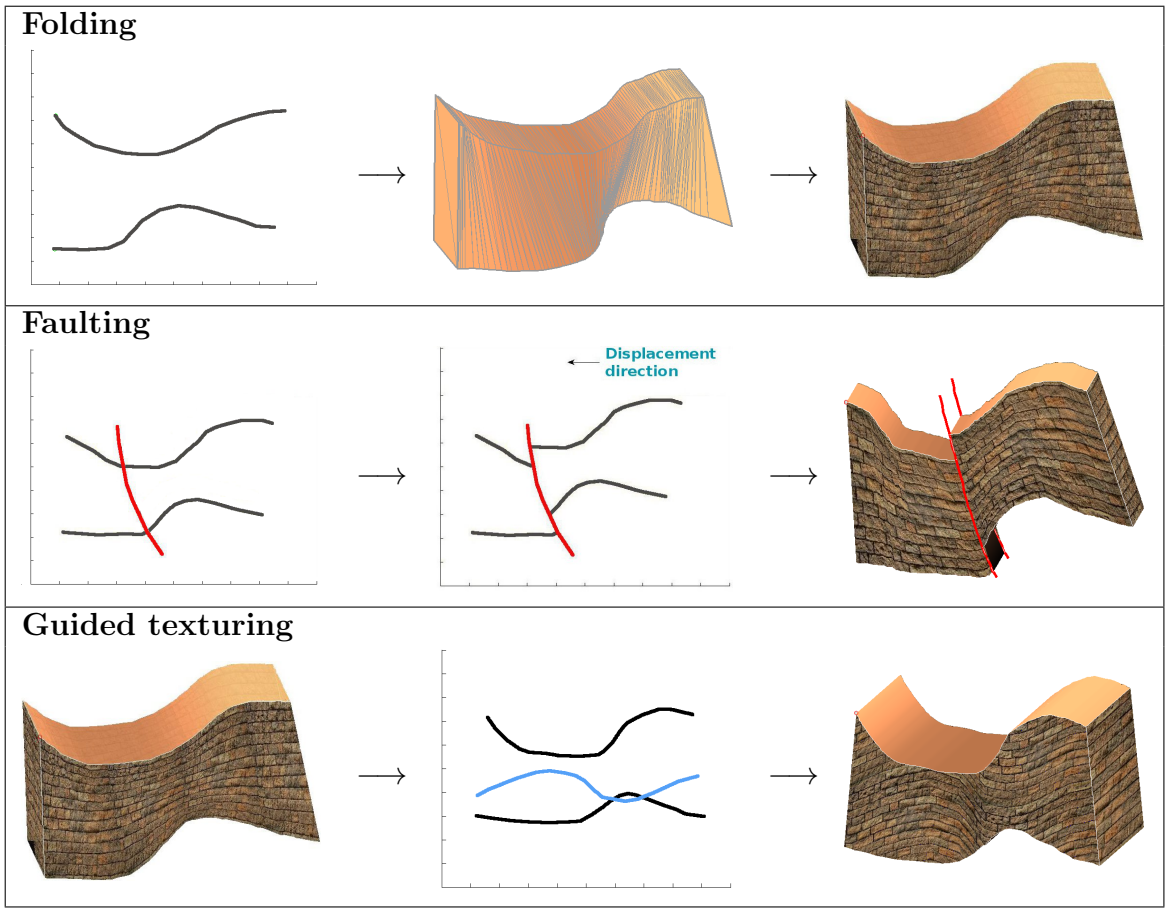


Figure 3.1: Global overview: (top) folding definition with just two strokes that generate the model, which is triangulated and textured on the side; (middle) faulting process after having specified the (red) fault and the direction of forces acting on it; (bottom) guided texturing, changing texture mapping from the default setup by a user-defined stroke (blue).

shape correction is achieved with a sketch-based user guidance, depending on the kind of process to be visualized.

If we consider the two green sketches b_1 and b_2 of Figure 3.2, we can suppose they define one of the layers of an illustration. Our approach automatically maps a chosen texture (in practice, a 2D rectangular image) to the planar region in 3D space delimited by b_1 and b_2. By default, the top and bottom edge of the source texture is mapped to the top and bottom curve of the target layer. Then texture coordinates are assigned to the interior of the layers using conformal mapping which minimizes interior angular deformation. With this procedure, the input texture adapts to the shape of the layer, as in the top-right image of Figure 3.1. In case the user wants the layer texture to deform differently, she can sketch two target curves (s_1 and s_2 in Figure 3.2) differently from the layer borders. A new mapping is then calculated and the texture is cropped to the layer area.

Figure 3.2: Texture obtained on the basis of a user's stroke ($\mathbf{s}_1$ and $\mathbf{s}_2$). $\mathbf{b}_1$ and $\mathbf{b}_2$ are the top and bottom boundary of the layer, whereas $\mathbf{s}_1$ and $\mathbf{s}_2$ are based on the user's stroke that is acquired to guide the behaviour of the texture.

3.3 Illustrative Animated Storytelling

In Paper 2, we extend the method described in Paper A to support animations. The idea is to employ the method described by Lidal et al. [71] to introduce a temporal aspect. In that paper, they emphasize the fact that, in many situations, communicating changes over time is fundamental when interpreting a geological process. They also propose a system that manages alternative interpretations of the same observed structural configuration.

In the same way 2D sketches tell geological stories that then can be compared to discuss feasible alternatives of an event (previous work by Lidal et al. [71]), we can, in Paper 2, automatically reproduce a geological animation that itself tells a story.

An animation is defined by means of sketched key-frame configurations. Intermediate time steps are obtained by interpolating the key-frame curves. Each 2D time step is extruded to a 3D model using the technique defined in Paper A. The conversion is implemented in Matlab and results in an animated film clip which take in the order of minutes to calculate.

When creating animations, the user must separately define the "shape morphing" and the "texture morphing", as the latter results in different meanings (see Figure 3.3).

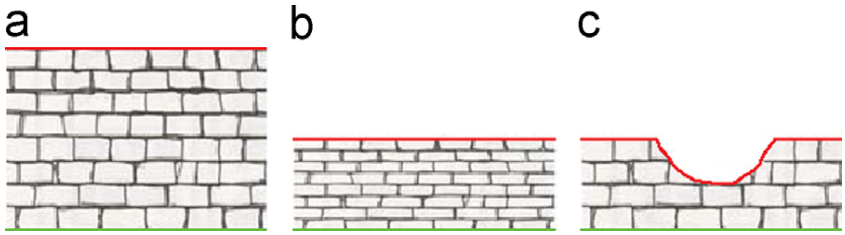


Figure 3.3: (a) Texture on the initial configuration of a layer; (b) compacted layer which is communicated by a compacted texture; (c) eroded layer which is communicated by a cut texture, where no texture deformation is performed. Image courtesy of Endre Lidal [73].

3.4 Layered Data Representation for Morphological Evolution

In Paper B, we give priority to layered structures and develop a representation for that, together with an accompanying volume visualization. The representation of the 3D layer-cake is encoded into heightmaps and volumetric visualization is achieved by ray-casting across the set of heightmaps. All features in a scene are introduced by means of sketched curves on the top surface of the model. Every curve is interpreted and converted to its corresponding heightmap. As shown in Figure 3.4, the interpretation of curves follows few simple rules:

- an open curve is a centreline of a tubular shape;
- a closed curve defines a deposit area when the user assigns a positive height, whereas it defines an erosion area when the user assigns a negative height.

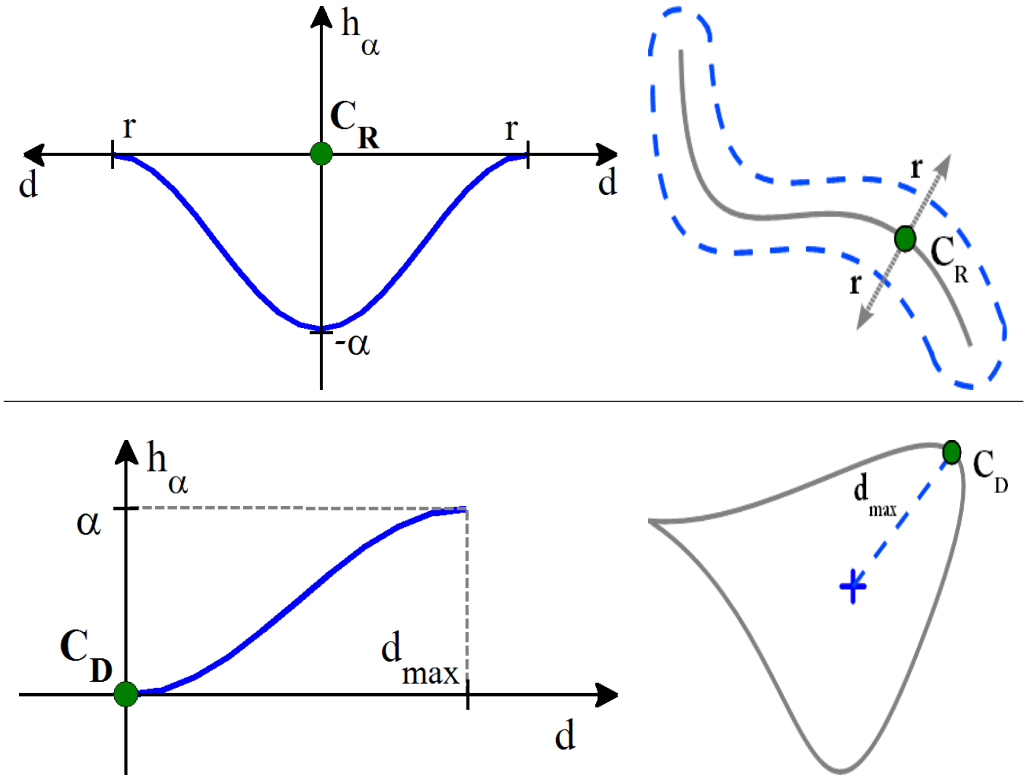


Figure 3.4: Top: the function h_α representing the section of erosion induced by the centreline. C_R is an arbitrary point along the centreline. Bottom: the function h_α defining deposition within a contour. C_D is an arbitrary point along the contour.

In addition, if enabled, interpolation between two curves (either open or closed) facilitates the process of defining a sequence of intermediate stages, that can be interpreted as different time steps of a shape evolution, as shown in Figure 3.5. The user only provides the number of intermediate time steps together with the initial and the final configuration.

From a centreline curve, we compute its region of influence. Within this region, a function that returns a height value for each grid point is defined. The value is inversely proportional to the distance to the closest centreline segment. When a point of the

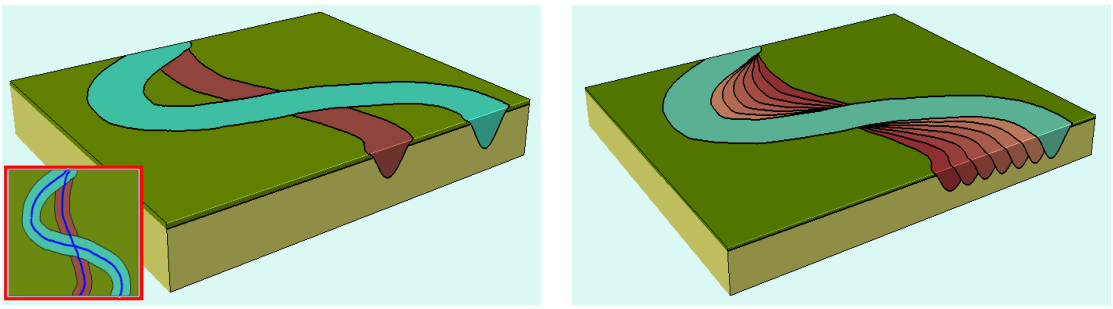


Figure 3.5: River evolution example. Left: first and last configuration of the river are sketched and imprinted. Right: imprint of additional five interpolated stages of the depositional history.

heightmap grid is outside the neighbourhood defined by the centreline, the height is set to zero. Similarly, we need to compute the distance field of the heightmap with respect to a closed curve to get the value of erosion or deposition inside the curve.

As defined in Paper B, relative and absolute layers (see Figure 3.6) are simultaneously kept updated and used during the modelling part (mainly relative) or during the volume visualization (absolute). Basically, they are both heightmaps, the difference lies in the values assigned to the grid points. When we refer to a relative layer, we know that the value at a grid point corresponds to the layer thickness. Therefore it is independent of previous events. The actual height value is instead dependent on the thickness of the layers below. When we refer to an absolute layer, we interpret the value at a grid point as the height of the layer at that point.

An identification number is assigned to each layer, and this number can be associated to a specific colour and volumetric translucency, reflecting specific properties of the layer.

The volume rendering takes place inside a user defined bounding box. For each fragment on the front faces of the bounding box, a fragment shader accumulates the colour along the ray, starting at the front face into the model until the back face of the bounding box is reached. Shading is performed at layer boundaries. When the layer id of the current sample is different from the layer id of the previous sample, we have crossed a boundary. The normal of the crossed boundary is found by calculating the central difference, looking up four samples in the corresponding layer heightmap. For samples that reside on the intersection of the bounding box with the solid model, we also check if the sample above, below, to the left or to the right have a different layer index. If so, the sample is on a layer boundary and we colour it black. This results in a layer separation line as shown in Figure 3.5, also found in hand made illustrations.

3.5 Model Discontinuities and Interactive Illustrations

Using the layered data representation that we introduce in Paper B, we propose an extension of it to represent layer discontinuities in Paper C. Our illustrations result in having independent layer blocks, produced by user-defined discontinuities, that can be displaced interactively with a simple slider. By layer discontinuity, we mean a surface $\mathcal{F}$ that separates the layer into two parts which are then displaced along $\mathcal{F}$. This surface defines where the initial heightmap is cut and how to build the two new heightmaps which encode the displacement.

When $\mathcal{F}$ intersects a layer, the layer is split and saved as two distinct heightmaps.

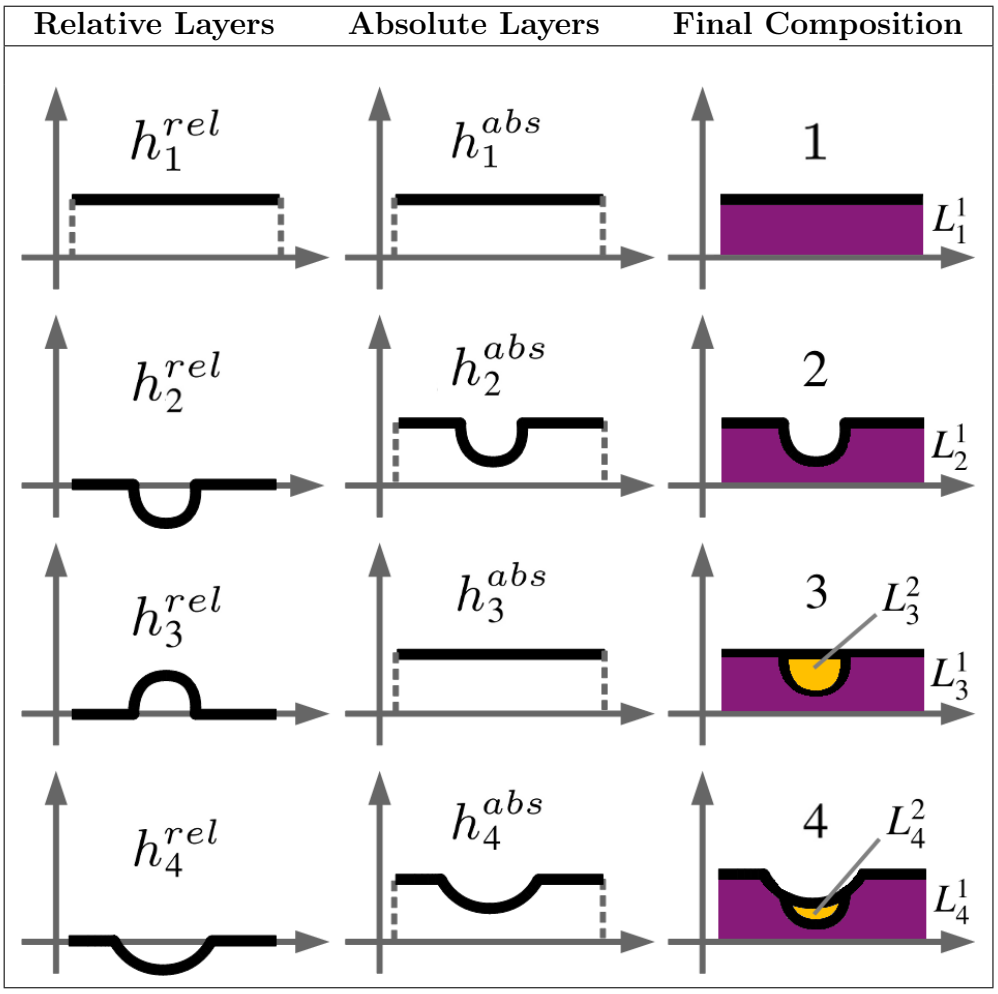


Figure 3.6: Relation between relative layers, left column and absolute layers, middle column. The right column shows the accumulated final model. From top to bottom is shown the adding of new layers, where layers 2 and 4 produce erosion (removal of material).

The displacement that is caused by the discontinuity is defined by the user and assigned to the layers intersected by $\mathcal{F}$ (only one of the two layers obtained by the splitting moves according to the displacement). A layer delimited by a top surface $\mathcal{T}$ and a bottom surface $\mathcal{B}$ that has a discontinuity defined by a surface $\mathcal{F}$, as shown in Figure 3.7, is separated into two parts. Their absolute values are given by

$$h_1^{abs}(i,j) = \begin{cases} h_{\mathcal{B}}^{abs}(i,j) & \text{if } j \leq j_Q \\ h_{\mathcal{F}}^{abs}(i,j) & \text{if } j_Q < j \leq j_P \\ h_{\mathcal{T}}^{abs}(i,j) & \text{if } j > j_P \end{cases}$$

and $h_2^{abs} = h_{\mathcal{T}}^{abs}$.

The process of faulting a model requires the use of both absolute and relative layers, which were introduced in Paper B. For a conversion from one representation to the other, we calculate new values for each grid point of the heightmap in parallel using the GPU. This results in a quicker procedure, allowing interactive animations.

Finally, we suggest and give examples of several visualization techniques that ease the internal inspection of an illustration. In Paper C, we show examples of cut-view, staircase view, exploded view and time-defined view, i.e. where we select a particular

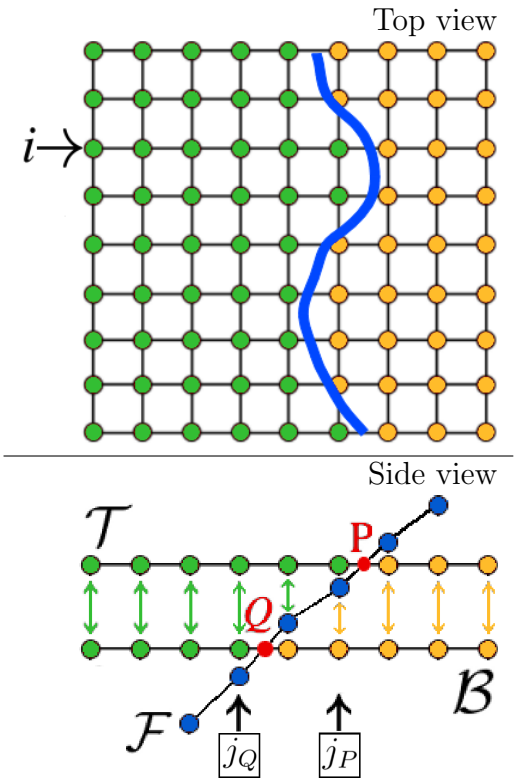


Figure 3.7: Top image: grid of the layer that is traversed by the discontinuity $\mathcal{F}$ (blue curve). Bottom image: cross-sectional view corresponding to the i-th row, where $\mathcal{F}$ intersects the grids $\mathcal{T}$ and $\mathcal{B}$.

time period to display by means of a temporal slider that goes through the time steps of the modelling process.

Chapter 4

Results

This chapter summarizes the principal applications and results obtained with the methods introduced in the papers. Our illustrations have application in geology as communicative support in interactive discussions amongst domain experts, for management decisions and for teaching purposes.

4.1 Folds, Faults and Erosion

Several scenarios may be explored in a meeting with domain experts. One possibility is that the topic is centred on the observation of different rocks and their intrinsic properties. Furthermore, a rock deformation given by folding and faulting processes is studied and discussed. To this aim we contribute with the work in Paper A, where sketches cover the need for folds and faults in the layer-cake, while deformed textures of particular rock types can be assigned to the layers.

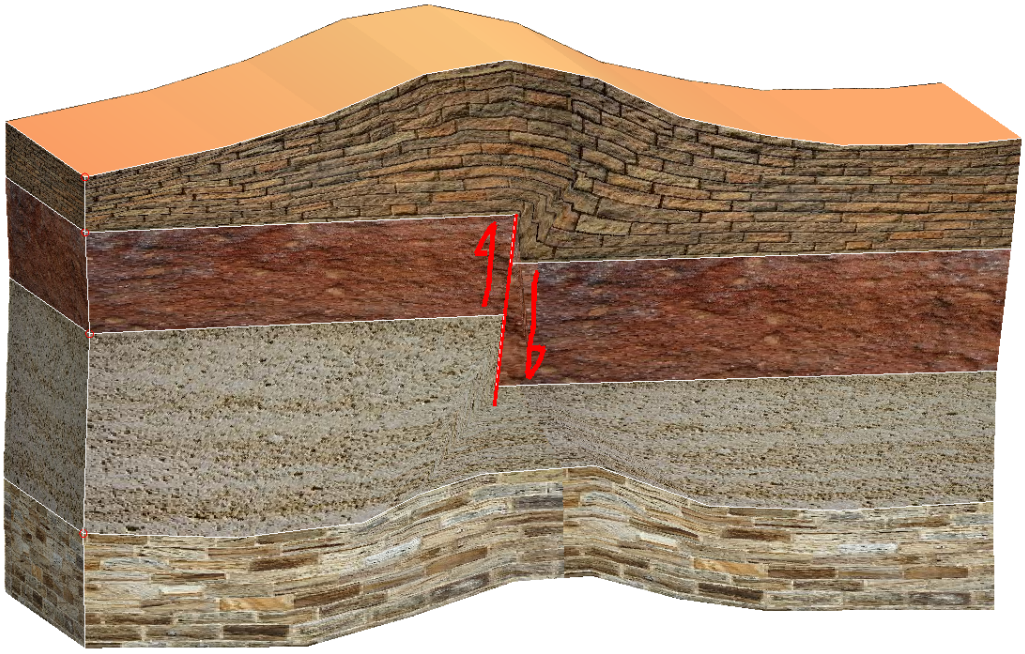


Figure 4.1: Layer-cake illustration obtained with our approach.

For the work presented in Paper A, we studied what information geological illustra-

tions share and salient points they have in common. We also discussed with geologists how they would have proceeded in sketching illustrations. We extrapolated common sketching procedures to achieve intuitiveness in our tool. After we implemented our prototype, we generated a few models, such as the one in Figure 4.1, to show to domain experts. Finally, we asked them to evaluate the speed of our approach.

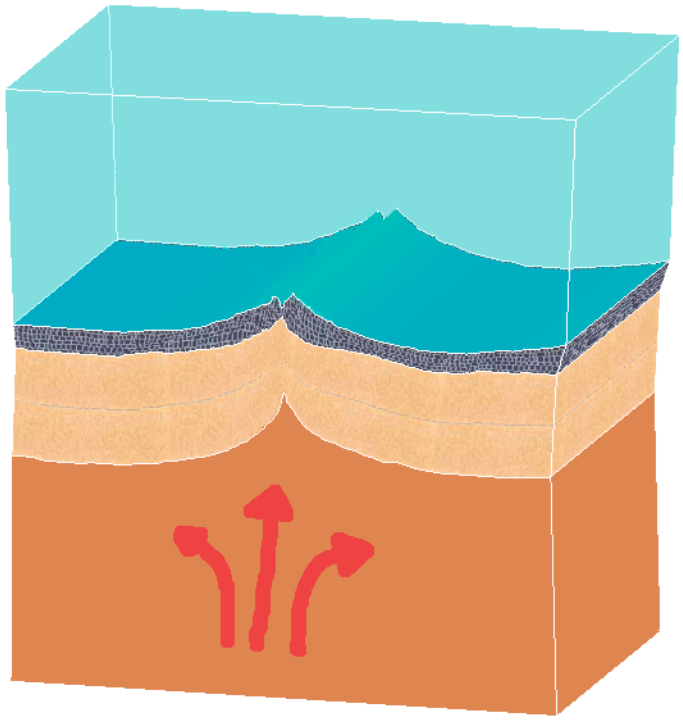


Figure 4.2: Example of the oceanic crust with red arrows suggesting movements of the mantle.

A comparison of times concerning Figure 4.1 and Figure 4.2 is given in Table 4.1, where we place side by side the estimated time used by a geologic illustrator to create a 2D illustration (rightmost column) and the time used to create a 3D model of the same illustration with our approach (second and third column).

Time	User interaction	Processing	Illustrator's estimation
Figure 4.1	$\sim$ 20 sec.	$\sim$ 2 min.	$\sim$ full day
Figure 4.2	$\sim$ 20 sec.	$\sim$ 1 min.	$<$ 1 hour

Table 4.1: Approximate comparison of time using our approach versus manual drawing.

Figure 4.2 demonstrates how to enrich illustrations with simple refinements that help to assign a context which is important to convey scales, environment or other useful information. For instance, layer transparency makes the top layer resemble water and projected drawings, used for notations on the model, show the pressure coming from the mantle.

Introducing animations in our illustrations enables us to present more types of processes. We are able to show geological events such as the forming of a *graben fault*, as in the example of Figure 4.3. In the left column, we sketched the initial and the final time step of the process. Intermediate key-frames are interpolated, extruded in 3D and rendered with the same texture as in the right column of Figure 4.3.

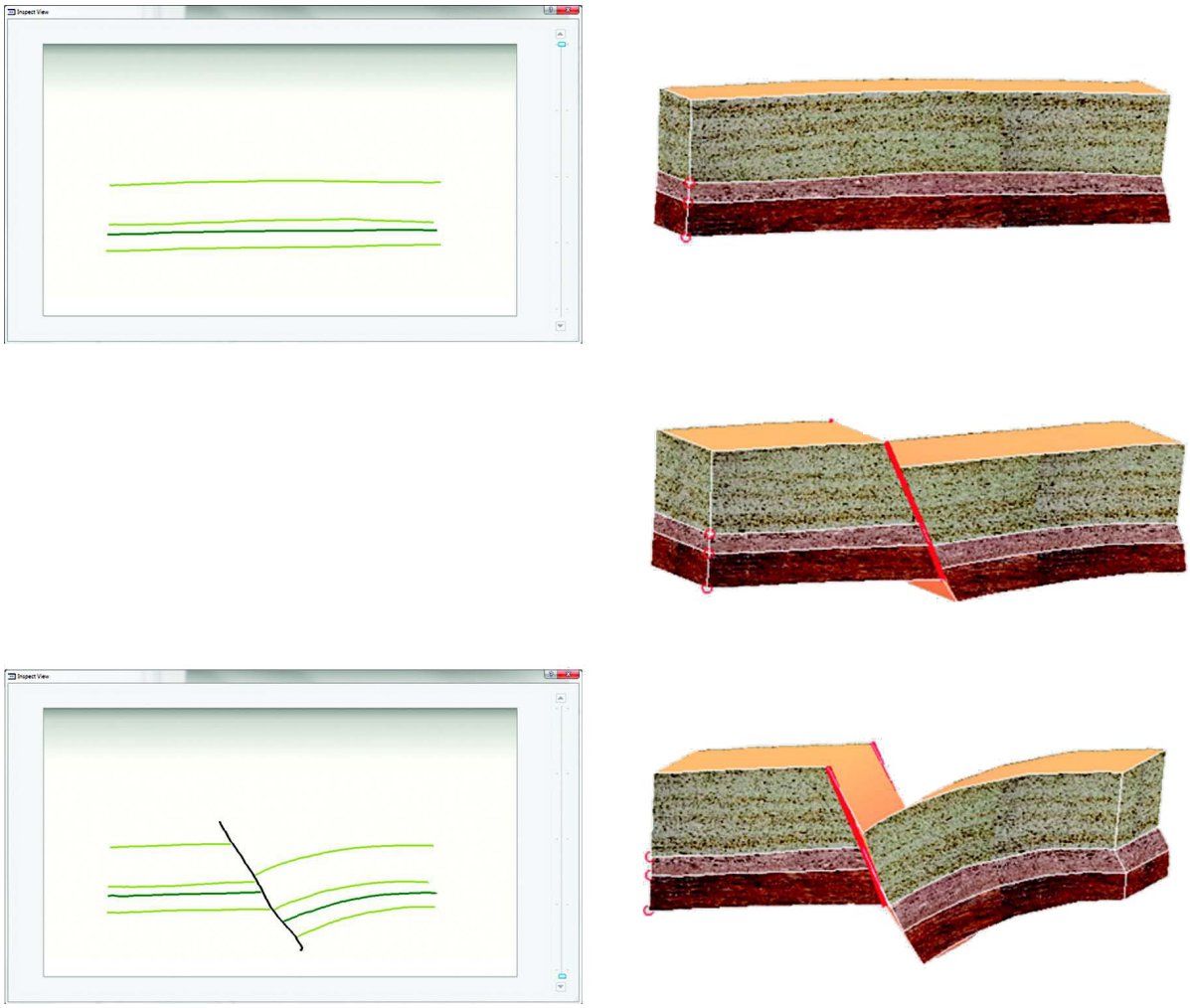


Figure 4.3: Still images of an illustrative animation (right column) and the corresponding initial and final sketched configurations of the geological event (left column). Image courtesy of Endre Lidal [73].

4.2 Stratification and Fluvial Systems

Internal architecture is important in the sense that it tells how a depositional element (e.g. channel or delta) evolved. Our approach offers a new way of producing illustrations by performing interactive erosion and deposition that lets the illustrator mimic processes that she interprets to have been the cause for fluvial system morphology.

One of the interests of geologists are ancient rivers, as they are possibly carriers of hydrocarbons, because rivers had flora located nearby and were visited by fauna. This can cause massive deposits of biomass to accumulate and become buried by successive depositions.

Models of basin stratigraphy, generated by fluvial system development, highlight sub-surface heterogeneity for natural resource exploitation. Furthermore, they can also be employed in the context of palaeogeomorphology, which is a branch of geology that studies ancient erosion surfaces.

A hypothetical meeting concerning a fluvial system development, as depicted in Figure 4.4, would host an expert on river and delta geology. The domain expert would, in short time, sketch the 3D model displayed in Figure 4.4 and, in the meanwhile, tell

her story about the formation process and how the meandering river would eventually deposit its transported material particles.

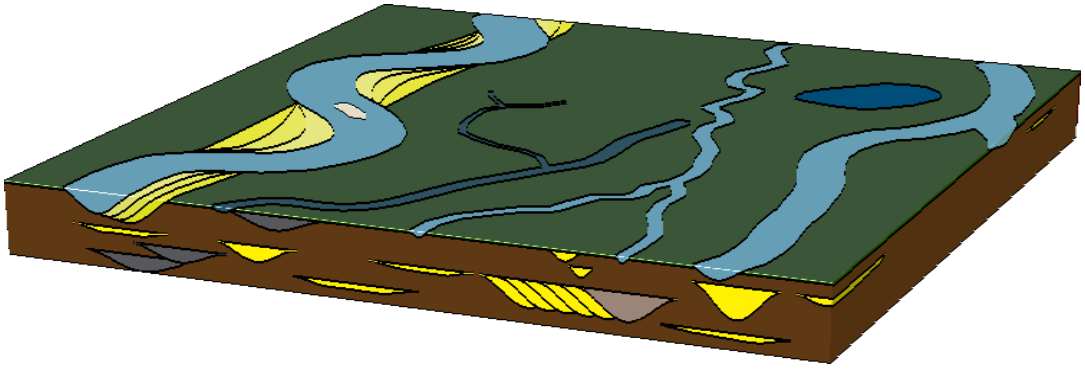


Figure 4.4: This example illustrates a real case analysis from field observations of a river sedimentation process [66]. It was recreated with our technique described in Paper B.

Paper B addresses sketching illustrations of fluvial systems. Their depositional history is important for geologists, because it allows them to detect sandstone formations. Evolution of rivers is characterized by many phenomena that can be considered or addressed in the modelling stage. For instance, there can be an interest in studying river braids (e.g. how and where, along its path, this occurs) to understand the reasons and consequences. One could also focus on the evolution of the shape of a river itself (e.g. curvature of oxbows or width of flow) and on how a river is influenced by, or changes the environment around it. Similarly, there can be an interest in the sedimentation process that is involved in a fluvial system [32]. That is, to model erosion and deposition of the internal and external bank of a meandering river, as well as *clinoforms*, which are subaqueous landforms generated by a delta deposition.

Geological feature are defined by sketching on top of the layers, as shown in Figure 4.5 bottom right.

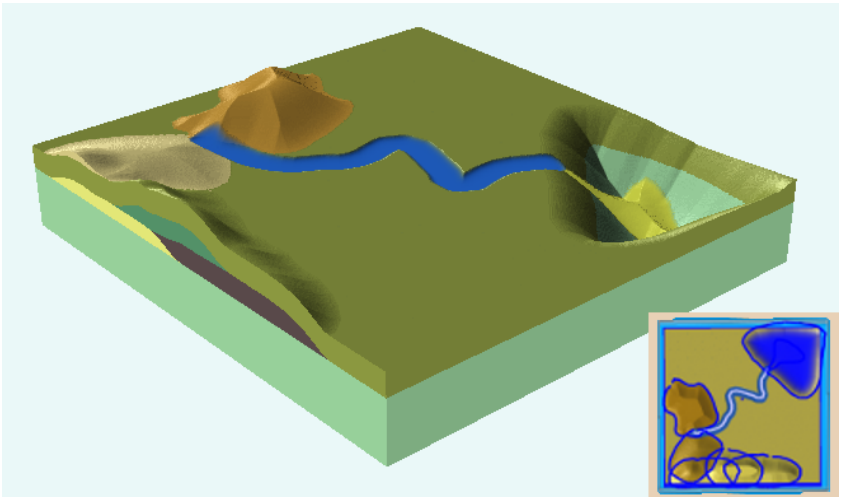


Figure 4.5: A 3D model created using our sketch-based approach to shape surface and subsurface geological features. Bottom right inset shows a map view of the sketched strokes in blue overlaid on the model.

Sketching Rivers. An open curve is interpreted by our system as a river centreline. A river evolution is sketched by defining a start and an end river in addition to specifying how many intermediate rivers the system should create. Each sketch has a user-defined height and width factor associated. For rivers, this defines the depth of the river erosion and the region of influence.

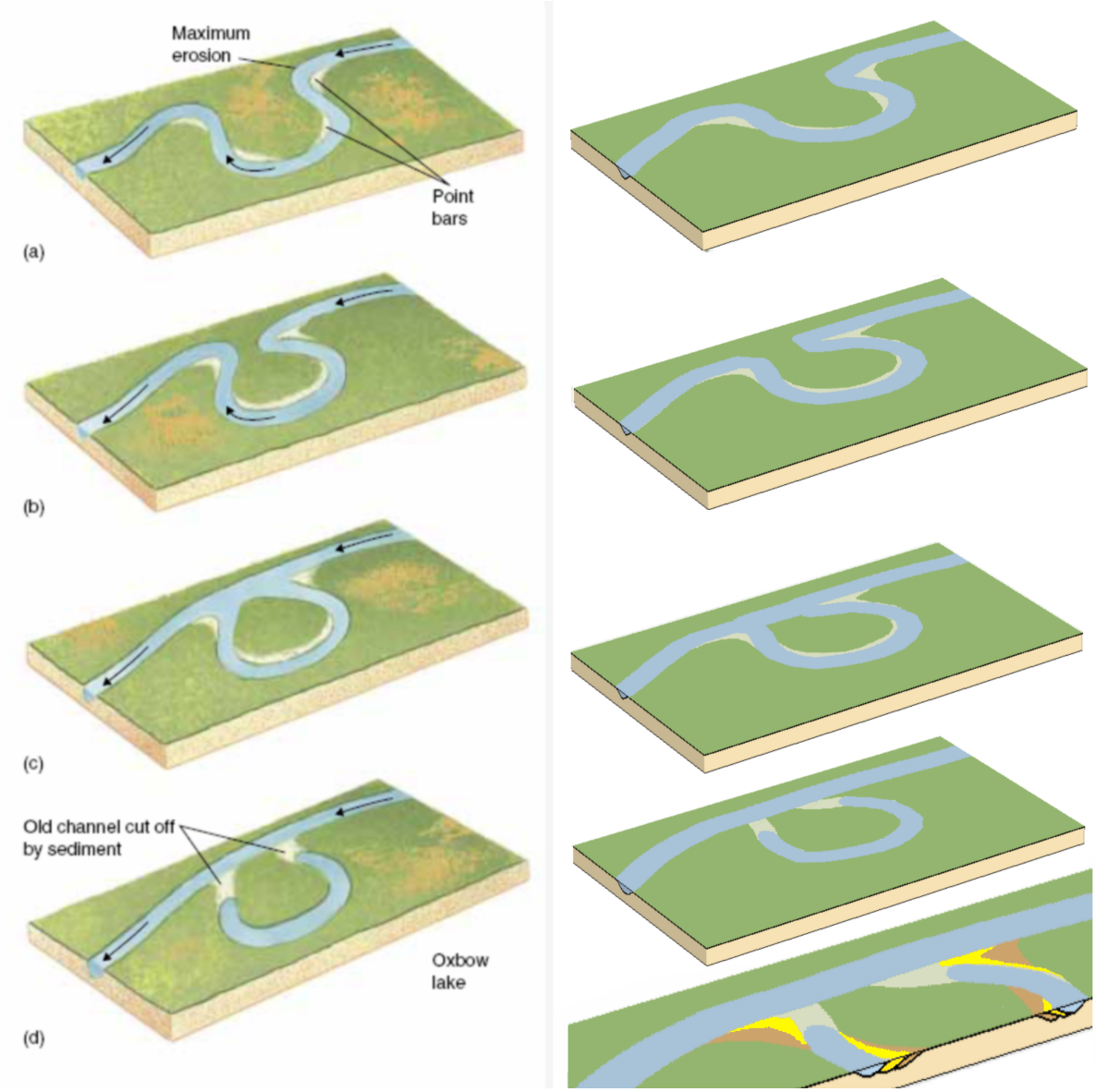


Figure 4.6: Left: sequence of illustrations showing the evolution of a meandering river with oxbow shape by Thompson and Turk [131]. Right: Reproduction with our approach requiring only five strokes. In the enhanced image at the bottom, we added colours to the individual depositions to show them separately.

Sketching Deltas. When sea level is sinking, deltas and shoreface deposits are arranged in successive, seaward-stepping sets, distinguished from each other by brief periods of altered depositional patterns (e.g., storms or sediment supply). Internally, these sets typically show coarsening upwards of grain sizes and therefore high quality reservoir sandstone can be found in the upper part of these sets. Similar effects with opposite

trends take place when sea is rising. Due to the successive deposits at varying positions, deltas have a complex internal structure which are tedious to model when defining each layer boundary individually. We propose a simple yet powerful sketching metaphor for modelling deltas including the varying grain size properties of each individual layer. A closed curve defines an area within which the delta deposits.

Sketching Mountains, Lakes, Constant Layers and Sea. Mountains, which are a source for depositional material, can be coarsely sketched with our approach by using the same sketching metaphor as for sketching deltas. This is not geologically correct, because mountains are usually not created directly from sedimentation. However from a modelling point of view, our sketching operators support expressing mountains in this way. Lakes can also be sketched for creating landscape features. Layers of constant thickness can be added to represent a base layer, a top soil layer or a series of layers of different material composition that can be later eroded for modelling outcrops. In addition, the user can define a global sea level where translucent volumetric water will be rendered for describing the subsea volume in which delta deposits are produced.

A property that is appreciated by domain experts is that our 3D sketched models can be easily internally inspected with cutting planes that enable multiple cross-section visualizations. This helps in understanding complex internal layering within the sandstone, otherwise not intuitively apprehensible (Bridge [17]). Sequentially defined models enable interactive discussions, fast hypothesis testing and creation of time-series illustrations, such as the one in Figure 4.6, that demonstrates how to achieve an illustration with our system (right) expressing the same river evolution process as in the manually made illustration to the left.

Our proposed data structure, and the way it is processed to render volumetric models, has advantages compared to a voxel representation. For example it has higher resolution with less storage requirements. This is due to bilinear interpolation of adjacent height values of the grids we use. The difference in resolution can be noticed in Figure 4.7, where voxel artefacts are clearly visible in the zoom in of the left image. When using a volume for storing individual segmentation masks, interpolation between segmentation id's make no sense and must be turned off. This results in staircase artefacts.

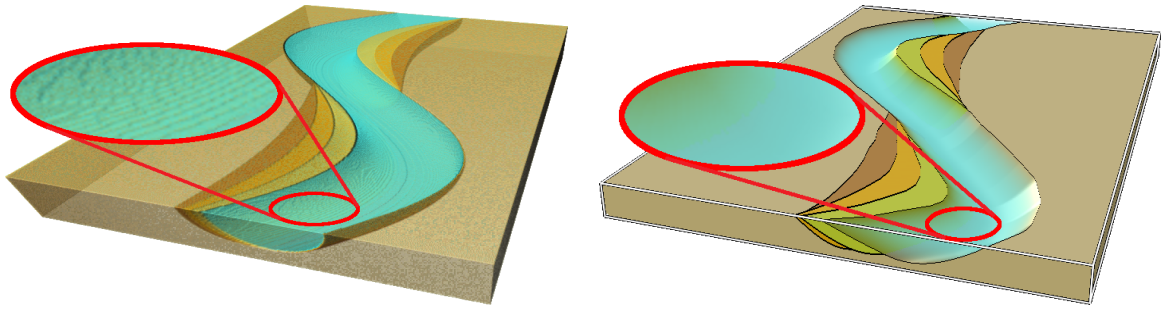


Figure 4.7: In this figure, we show the improvement in resolution from using a voxelization of the model (left image) to using our volume visualization obtained by ray-casting through heightmaps (right image).

Geological features are defined through sketches. Each sketch can generate deposition and erosion processes; in particular, rivers and channels, mountains, basins, deltas and intermediate stages of their evolution. This latter processes of delta deposition and

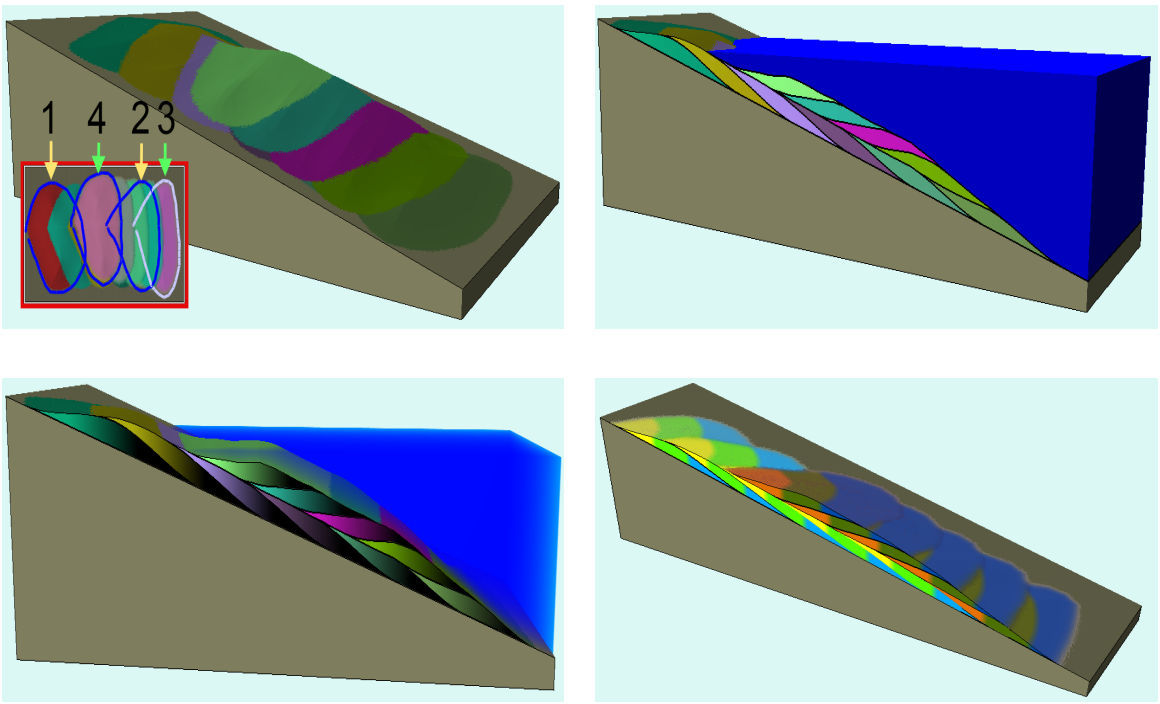


Figure 4.8: Delta stratigraphy example, where only four sketches define the model: sketch 1 and 2 describe the sedimentation process that moves from the shoreline towards the sea; the second sedimentation process accumulates deposits in the opposite direction and is defined by sketch 3 and 4. The two lower images also show different grain sizes for each deposit.

development in time is shown in Figure 4.8, in which more phases of deposition are automatically added by interpolating two curves. Few strokes define the whole model and the cut-views highlight the grain size distribution encoded in the delta deposit colour map.

Grain size of delta depositions is dependent on the distance to the mouth of the river as heavy particles deposit first. Our system allows the user to define this feature and to visually convey it, as represented by Figure 4.8 in the two lower images. The example in Figure 4.8 has been created with just four sketches: the first (in chronological order) sedimentation process goes towards the sea and is defined by sketch 1 and 2 (interpolated with three intermediate steps); the following sedimentation process proceeds towards the shoreline and is defined by sketch 3 and 4 (interpolated with three intermediate steps). Grain size can be expressed by darkening layer colours, as in the bottom-left image of Figure 4.8, or by letting the user utilize customizable colours for each of the depositional layers, as in the bottom-right image of Figure 4.8, where we define three colours for each deposit and, in addition, we chromatically distinguish between the first sedimentation process and the second. As is natural for illustrations, Figure 4.8 is an over-simplification of the complexity of a real situation.

Evaluation. Our prototype is implemented in *Volumeshop* [19]. See Figure 4.9 for a snapshot of the graphical user interface we have implemented. We used our prototype to generate a tutorial video [88] that was shown to geoscientists with different expertises for a user evaluation. In addition, other domain experts directly tried our tool before giving us feedback.

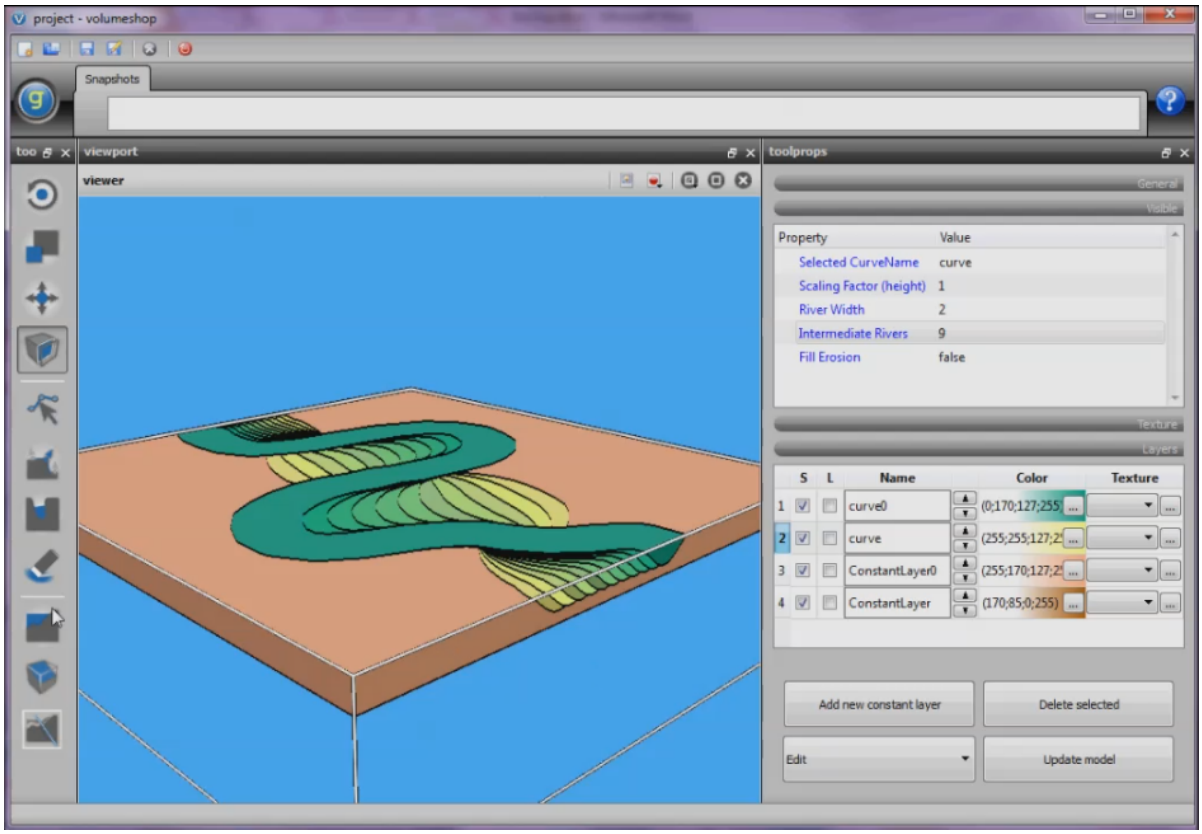


Figure 4.9: Image of the graphical user interface which was employed for the user study and to create our examples in Paper B and Paper C.

Details of the results and discussion on the user study can be found in Paper B. Here we list some of the comments we received by the participants:

- "The system seems really useful if you study fluvial systems behaving in a regular way."
- "There is a lot of potential in a piece of software like this. In addition to help visualizing systems in 3D, it could speed up the process of creating illustrations."
- "Great job indeed!"
- "It seems like a nice and easy modelling tool."
- "The program has good potential and can be of great help in visualizing in 3D."
- "The program definitely has good potential. As geologists often deal with 2D outcrop sections and build a picture from many pieces, this program can really be a powerful tool in 3D imaging of geological processes."

4.3 Interactive Faults and Compaction

The process of faulting is, in general, restricted within the first 15 kilometres of the Earth's crust. Faults are important geological features. Their interpretation leads to an understanding of the behaviour of the crust of the Earth. Movements in the crust

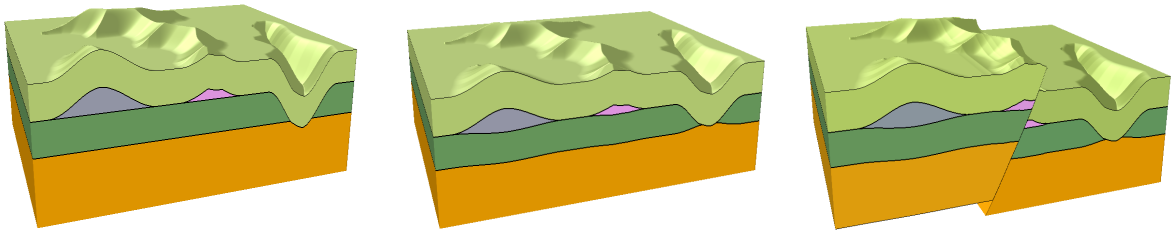


Figure 4.10: To the initial stratigraphic illustration (image to the left) we firstly introduce compaction (middle image) and then we add a fault (right image).

produce faults in rock layers. For instance, a standard approach to derive the direction of two lithospheric plates is to study faults generated by their displacement. A clear manifestation of active faulting is given by earthquakes. Studies on active faults cannot foresee earthquakes, but they can give hints on where to expect seismic activity. Moreover, understanding the types and patterns of faults have other implications on the prediction and reconstruction of ancient landforms. This is because different types of faults tend to form in different scenarios, and, for example, faults at active rifts are different from those along edges of mountain ranges. Another important consequence of faults is that they can change the movement of groundwater, generating a strong influence on the distribution of mineralisation and also on the subsurface accumulation of hydrocarbons.

Compaction contributes to the formation of sedimentary rock by squeezing out air and fluids that exist between sediment particles. The squeezing process is due to the weight of overlying layers of sediment. Faulting and earthquakes shatter rocks that are then compacted to form new aggregate materials, sometimes of economic value.

This section of the thesis is focused on achieving illustrative geological models showing non-planar faults and compaction of layers. Paper C describes in details the steps that are needed to obtain our goal.

Going back to the hypothetical meeting, a geologist which is expert in faulting and compaction takes the word and explains to the audience why the compacted rock layer has yielded hydrocarbon leakage in its upper level. The leakage has gone through passages opened by a fault fracture all the way up, until an impermeable layer has been encountered.

We have extended Paper B to include faults and compaction. We also transfer the most computational expensive tasks to the GPU by implementing computing intensive parts of the code in CUDA. This helps us to maintain interactivity during modelling of faults (which requires more computations than what was needed in Paper B for stratification processes). Parallelization is done on the points of the heightmap grid, therefore every procedure on the GPU consists of computing calculations with a low number of variables corresponding to the number of layers. With our implementation, we are able to quickly create models, such as the ones in Figure 4.10, using few sketches. Figure 4.10 shows a stratigraphic model to the left, a compacted version of the same illustration in the middle (the lower a layer is, the more compacted it is) and finally a fault in the right image. Notice that in the two right images of Figure 4.10, the orange bottom layer does not expand where there is erosion, as it might seem. There is just a relative effect of compaction around it.

Compaction of layers is given by a coefficient which defines material compressibility. The coefficient is dependent on the weight of the material deposited on top. When an

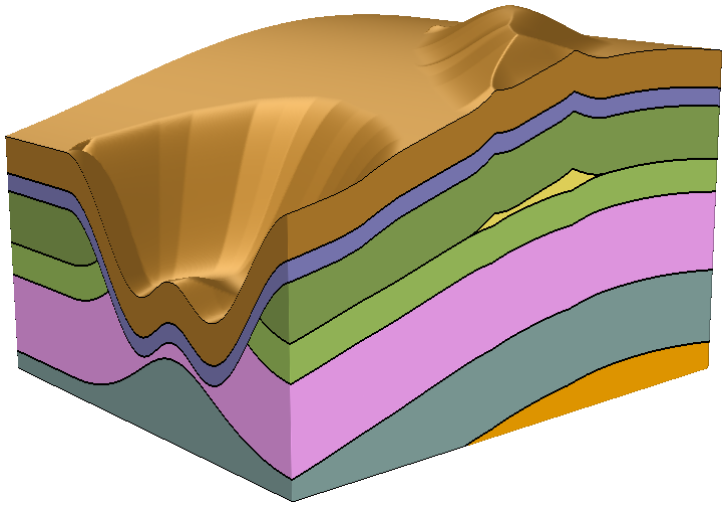


Figure 4.11: A negative coefficient of compression, due to erosion, generates expansion of a layer thickness as can be observed in the centre of the cavity.

erosion is encoded in one of the above heightmaps, we obtain the effect of expansion on the layer below. This effect allows us to obtain an initial stage of intrusion shapes in the illustration, as visible in the example of Figure 4.11.

Our approach differs from other previous techniques when considering the range of faults that can be modelled. Non-planar faults are important in geology and we support them by giving the user the possibility to directly draw the desired shape of the fault surface. Figure 4.12 displays an example of a non-planar fault, where some of the left layers have been set to be transparent to better show the actual internal shape of the fault surface. Although the fault is planar on the side in this example, this is not a restriction in our method.

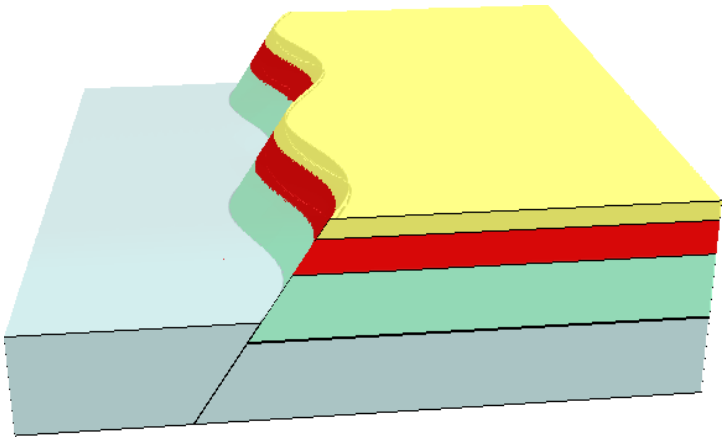


Figure 4.12: In this example we show a non-planar fault. The three top layers of the left block are set to be transparent to reveal the fault shape.

Chapter 5

Conclusions and Future Work

Tools for rapid geological modelling do not exist for subsurface applications, and the idea of rapid model updating and visualization may be an excellent teaching tool and dissemination medium. Our contribution provides this.

We present different ways of defining a 3D layer-cake. Our choice is dependent on what we want to focus on, i.e. the common denominator in our work is given by the exploitation of sketches to obtain intuitiveness and rapidity in modelling; but each geological process or set of geological processes has been treated by us with diverse approaches to best suit domain needs or wishes. For instance, in the first paper (Paper A), we mainly focus on faults and folds. Therefore it is important to be able to draw on a side (cross-section) of the model.

In Paper B, we concentrated our efforts on modelling and visualization of fluvial systems and their depositional history. In this case, sketching on top of the layers, from a top view, turns out to be more convenient for those who design an illustration.

To give the user full control of the modelling and to allow interactivity, we have not incorporated the aspect of physically based simulation of river evolution by simulating matter transport in water. Several existing works already suggest solutions to this.

Depositional processes of rivers and deltas implicitly define internal structures, which require a volumetric representation to display them. The same volumetric representation used in Paper B is employed in Paper C, where internal features are in addition fractured by faults and compacted by gravity.

Coulthard and Wan De Wiel [32] suggest landscape evolution models as a method for modelling river history. Following this line, we propose a layered representation where each layer is a height map of a certain time step.

The simple idea of interpreting height values of a relative layer as amount of deposition or erosion leads to an intuitive definition of a geological process and to basic and parallelizable arithmetic operations during the computations among layers. The various stages of deposition and erosion of a river or a delta are captured by the data structure and rapidly retrieved for visualization.

Geologists, arguing for a feasible interpretation of the stratigraphic configuration of the Barents Sea in the above mentioned meeting, benefit from the contributions that can be found in this thesis. More precisely, the following contributions cover their requirements, but are missing from previous work.

- A rapid sketching tool for creating illustrative visualizations of structural geology, as seen in geology text books which can be beneficial in exploration companies to describe subsurface situations, that includes:

- modelling of a faulting process of a rock layer through simple sketched curves;
- texture shape modification, in order to communicate different geological events, according to user guidance (in the form of strokes);
- application of conformal map for smooth texturing of deformed rock layers (to maintain the meaning of illustrative textures on rock layers);
- animation of 3D synthesized models, to convey geologic processes;
- a volumetric rendering algorithm based on a novel compact representation to interactively obtain illustrative layer-cake visualizations;
- fluvial systems with their history and evolution, together with a visualization of their depositional imprinting (useful to model ancient river channels which are present in the lower stratigraphic layers of the Barents Sea);
- customizable colour distribution to convey the variation of different properties of a deposited material such as grain sizes in a delta deposition;
- support for compaction of subsurface layers;
- support for non-planar faulting processes and their interactive animation;

Although our main application is in a geological environment and our tools target specific geological aspects, we see various fields which could benefit from interactive domain specific sketching. This is true anywhere where a 3D model can be used as an illustration of a process, and when there is a group of people that has to take a decision that is dependent on the configuration of the considered event. For instance, our approach could be adapted for application in archaeology, terrain inspection before laying the foundation during buildings construction, towards geothermal industry, ore extraction or aquifer detection.

Earth science disciplines are increasingly interested in modelling methodologies developed within the computer graphics research. And they are driven by the need for rapid modelling procedures. An interesting research direction can be the consideration of temporal aspects in geology, as investigated in Paper 2. Erosion has been considered in this context, but geological processes are driven by many more phenomena. Here, also the temporal aspect can benefit from user input in the form of sketched information.

Customizable river and delta sections is also a foreseeable extension of this work; the user could sketch the profile of the river section instead of having a fixed analytical function for that as we currently do.

A natural extension of this work could support co-rendering of the model together with an underlying seismic dataset or, in general, any measurements such as well logs or magnetic data, if they exist. In addition, it would be possible to import real landscape heightmaps (*Digital Elevation Models (DEMs)* are a widespread representation for that purpose) and combine them with user defined changes to the initial geomorphology.

Part II

Scientific Results

Paper A

Rapid Visualization of Geological Concepts

Mattia Natali[1], Ivan Viola[1,2], Daniel Patel[2,1]

[1]University of Bergen, Norway
[2]Christian Michelsen Research, Bergen, Norway

Abstract

We describe a sketch-based system for constructing an illustrative visualization of the subsurface. An intuitive and rapid modelling tool is defined, which takes as input user's strokes and creates a 3D layer-cake model of the earth. Our tool enables users to quickly express and communicate their ideas directly using a 3D model. For sketching, we have created geometric operators that capture the domain specific modelling requirements. We have devised sketching operators for expressing folding and faulting processes. This makes it possible to produce a large span of scenarios. Moreover, for communicating layer properties such as rock type and grain size, our system allows for associating user defined texture to each layer which can be deformed with a few sketch strokes.

This article was published in *Proceedings of XXV Conference on Graphics, Patterns and Images (SIBGRAPI 2012)*, pp. 150–157, DOI: http://doi.ieeecomputersociety.org/10.1109/SIBGRAPI.2012.29, and presented at SIBGRAPI 2012 in Ouro Preto, Brazil by Mattia Natali.

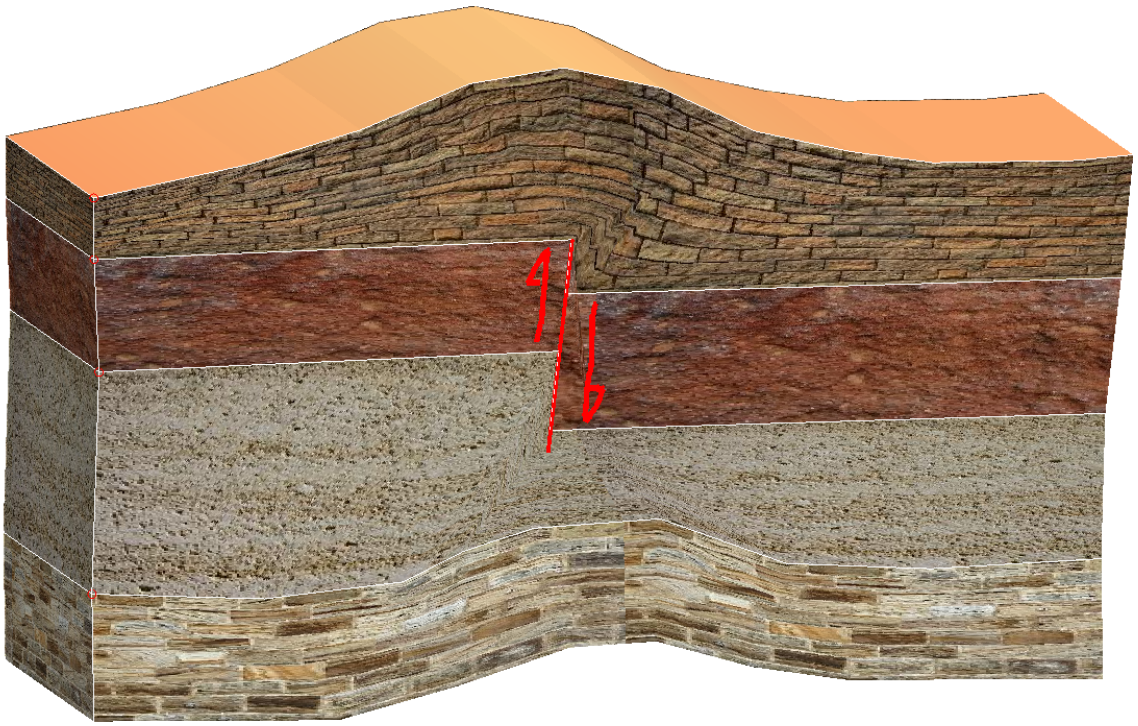


Figure A.1: Layer-cake illustration obtained with our approach.

A.1 Introduction

Many tools have recently been developed to permit better flow from human visual thoughts to digital representations. One approach in this respect is to use sketch-based methods. They give more freedom than *Computer Aided Design (CAD)* systems in designing a general shape. Another advantage of sketch-based methods is that, during planning, giving only a draft of a model is faster than providing all the shape details. In turn, this increases convergence towards the final structure. Detailed models are often made with tools that can be unintuitive for non-experts. The relatively new techniques of free-form sketching have found many applications: for example in toy generation (*Plushie* [85]), generic object deformation (*Teddy* [60], [42, 35, 26, 96, 94]) and terrain modelling [48, 14, 141].

It is advantageous when 3D digital models can be created without processes which require many hours of work. Procedural methods allow the generation of detailed models by only defining few parameters. But these types of tools lead to an approximation of the model that we want to create due to limited control of the outcome. Procedural methods do not give complete freedom in the construction. They give general global control, but no specific local shape control. A possible alternative is to use sketch-based techniques, replacing the use of parameters with free-form user strokes.

Our goal is to quickly create the 3D illustrative models found in geological text books. Figure A.1 shows an example made with our method. The produced model will be a way for geologists to express and explain processes involved in the subsurface. The tool could also accelerate the creation of geological illustrations in text books.

With our tool, the user first draws the boundaries of geological layers on a 2D vertical plane. Then, by sketching two of the most important geological processes, folding and

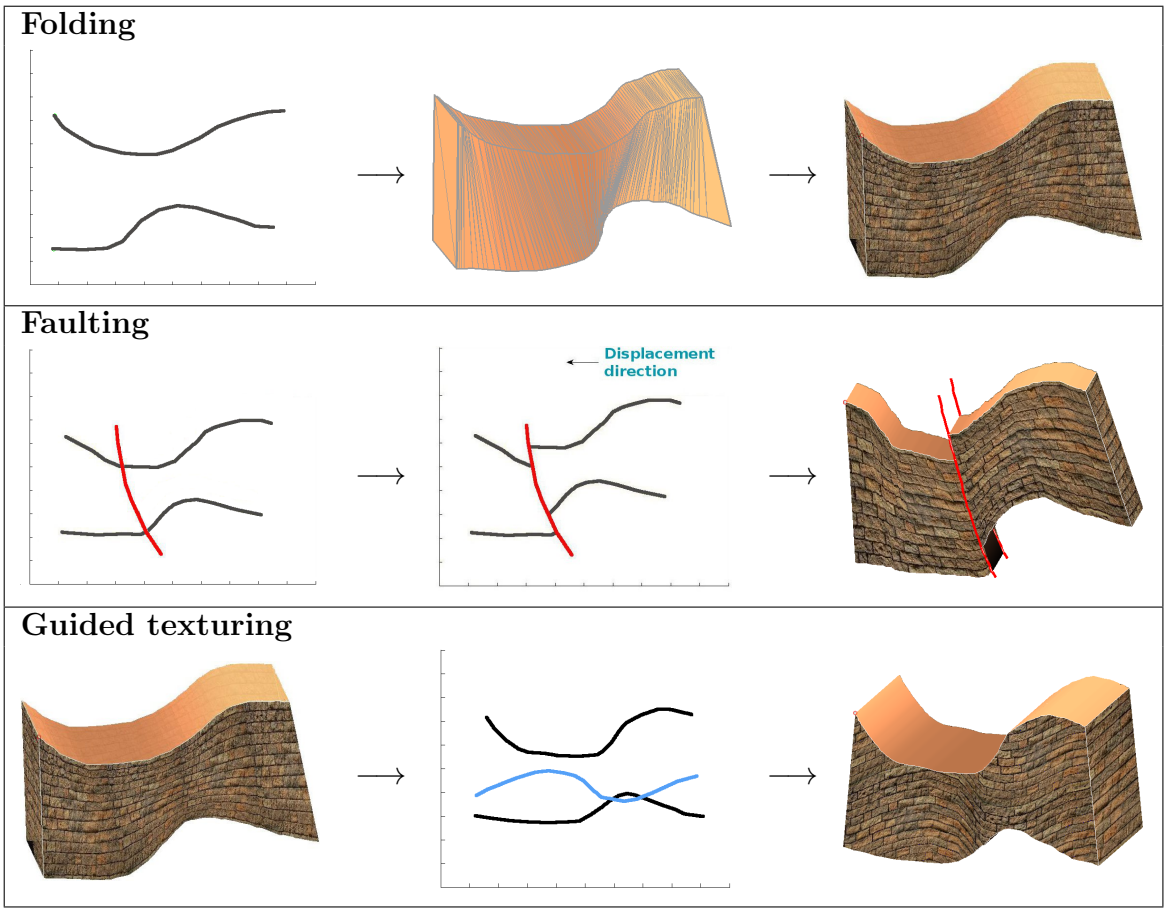


Figure A.2: Global overview: (top) folding definition with just two strokes that generate the model, which is triangulated and textured on the side; (middle) faulting process after having specified the (red) fault and the direction of forces acting on it; (bottom) guided texturing process, from the default adapting texture to the user-defined (with a blue stroke) behaviour of the texture.

faulting, we are able to illustrate a large amount of situations and behaviours occurring inside the crust of the earth. To further give a context to the model, textures are assigned to each layer and deformed according to the shape of the layer. The user can override the default texture deformation using sketching if it is not representative. The user-defined texture deformation can be used for expressing the processes of erosion and tilting. Figure A.2 shows how to define a fold, a fault and a guided texture.

Our method focuses on simplicity and speed as opposed to time consuming detail-editing of complex models. That is, we develop a method for producing fast qualitative results, since users lack such a tool in the geosciences.

The next section lists motivations for our research. In Section C.2, we give a short overview on related work on sketch-based modelling, for general modelling and for specific modelling in natural sciences. Section A.4 describes the type of representation we obtain and states the goal of the tool and what can be designed with it. Furthermore it gives a detailed description of the method. We demonstrate our technique in Section C.5 and give conclusions in Section C.6. Finally, we propose possible future work in Section A.7.

A.2 Motivation

Geologists need tools for generating earth surface and subsurface renderings in a rapid way. We introduce a method that is specifically adapted to create such geological models. During the development of our work, we target modelling problems encountered in geology, thus our method is not meant as a general modelling tool. For the latter purpose, many attempts have already been done. We achieve our scope by introducing geologically relevant sketching techniques, thereby getting an illustrative visualization of subsurface stratigraphy (material layering). The result is a qualitative representation giving a user the possibility to communicate how the earth has behaved or will behave, and of processes that take place. These illustrations make part of a geological illustrator's work, either in subsurface exploration companies, for showing expectations and results, or in text books writing, for visual explanation by expressive examples.

The work in this paper springs from requirements made by geologists, specified during meetings we had with them. They were important in the choice of the geological attributes we have introduced in the sketching toolbox. Several other geological attributes could be introduced in our layer-cake representation as geometric operators that would just need simple sketch-based input from the user. Amongst them we find sketching erosion, channels, salt domes that induce neighbour layers deformation, delta-shape to describe landscape changes due to river flowing, dikes or igneous intrusions. However, in this paper, we have focused on two fundamental phenomena taking place in the earth. They can originate everywhere, are important for interpreting earth movements and can be obtained with a few expressive operators. They are known as the process of *folding* and the process of *faulting*. A fold is obtained when elastic layers of rock are compressed. It is defined as a permanent deformation of an originally flat layer (usually produced by a sedimentation process), that has been bent by forces acting in the crust of the earth. The deformation generating a fold may have different origins: tectonic and convection stress; hydrostatic pressure; pore pressure; high temperature range; salt-, igneous- or sand-intrusion. Folds can have different sizes and occur either isolated or inside a set of deformed layers. Faulting, on the other hand, originates when forces that act on a specific layer are so strong that they overcome the rock's elasticity and yield a fracture.

The novelty of our paper lies in the combination of sketch-based modelling of stratigraphy together with the illustrative representation of geological features. In such a way, models are easy to create and simple to understand such as illustrations in text books, as shown in Figure A.3.

A.3 Related Work

As far as we know, no previous works discuss sketching subsurface geological models. Rather, there are some articles dealing with texturing stratigraphic models, like, for example, by Patel et al. [103], by Takayama et al. [129] and by Wang et al. [140]. All of them use volumetric textures for visualizing layered stratigraphy, but none of them focus on sketch-based, fast definition of the models and their appearance.

Recently, many surveys on sketch-based techniques have been made [33, 30, 95]. Some papers, related to *Sketch Based Interfaces and Modeling (SBIM)* have gained much attention such as *Plushie* [85]. *Plushie* introduces a method enabling 3D free-form sketching to create a rounded and smooth object that fits well as a toy prototyping application. Its pipeline consists of specifying two dimensional input strokes which define the silhouette

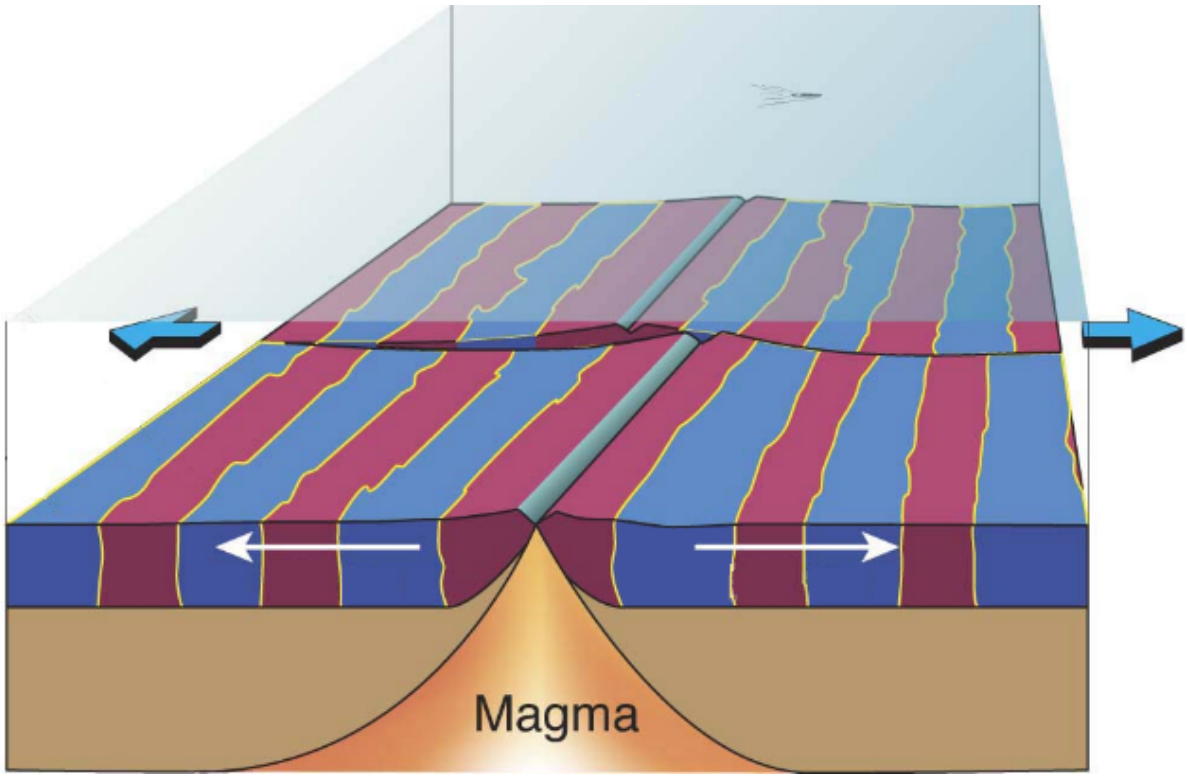


Figure A.3: Example of an illustrative image (courtesy of Haakon Fossen [46]) that can be found in a geological text book.

of the object from different points of view. Afterwards, they inflate the silhouettes to a 3D shape and allow the user to add other features directly on the 3D shape. Another method, presented by Brazil et al. [139], takes as input a set of strokes and constructs an implicit surface, interpolating the positions and normals of the samples. They use *Hermite Radial Basis Function (HRBF)* interpolation. The computationally expensive calculation of the implicit surface reduces interactivity when a large number of strokes occurs.

SketchUp [3], is a mature sketching tool, primarily made for architecture. We compared our method with it [135] and we observed that subdividing a solid by drawing folds in a 2D plane is quite similar to our approach. However, *SketchUp* takes more time to reach a final geological model. Moreover, it is not straightforward to generate displacements along partial intersections for modelling blind thrust faults (i.e. when the the top layer is not faulted, such as in Figure A.4 and Figure A.5). In addition, a single fault can only be generated as a simple straight line that cuts the model [136]. While with our method, a fault is defined on the geological cross-section as a generic sketch together with a definition of the direction of stresses converging to or diverging from the fracture. In addition, free form deformations of textures is not possible in *SketchUp*. We allow such a guided texture deformation on surfaces (for solid texturing, Zhang et al. [145] already proposed their solution in 2010).

There is much focus on modelling terrains (one example is given by Peytavie et al.'s paper [107]), but few has been made exploiting the intuitivity of the sketch-based approach. One of the latest works which focuses on creating and deforming landscapes with *SBIM* techniques has been published in 2009 by Gain et al. [48]. They describe a procedural terrain generation tool named *Terrain Sketching*, with the support for

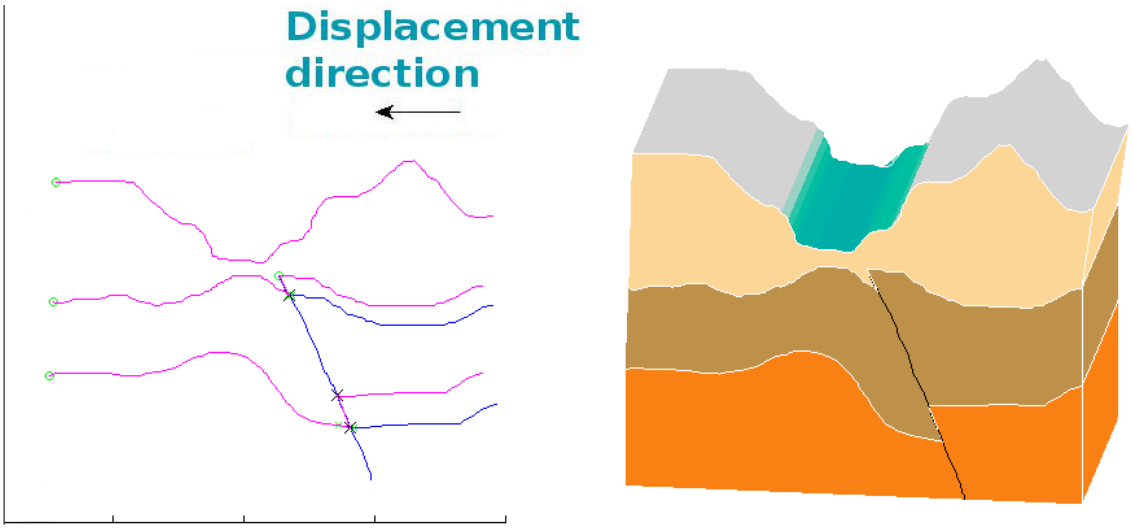


Figure A.4: On the left, three layer boundaries and a fault with its force have been defined. After extrusion, a river has been drawn to obtain the final model to the right.

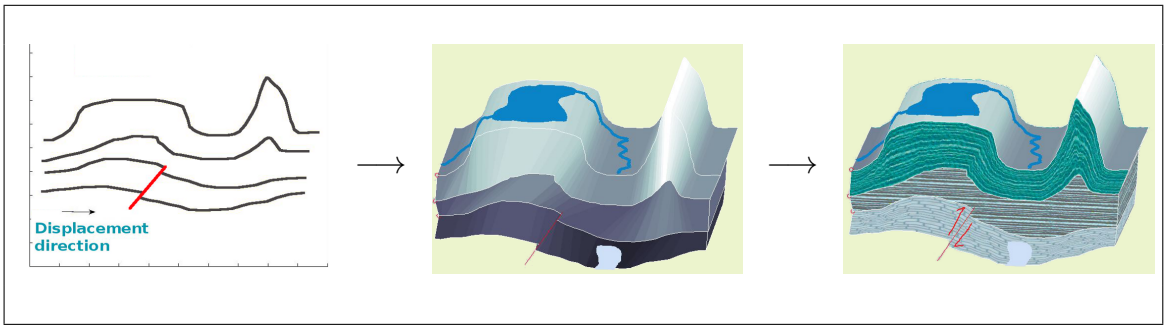


Figure A.5: Complete example of our method, from user's strokes to 3D illustration. Left image shows acquired sketches: black curves define folds and the red curve defines a fault. The model with geometry and projected drawings of a lake and a salt dome intrusion in the bottom layer is shown in the middle image. The final textured model is on the right, with guided texturing adopted for the middle layer.

sketching mountains and valleys. Their approach overcomes some limitations of previous methods on sketch-based terrain modelling: for instance, Cohen et al. [29] only allow straight areas of influence and boundaries; Watanabe and Igarashi also employ a straight shadow line and they do not give the user the possibility to change the proposed shape. Zhou et al. [147] allow landforms to have a more freely definable shape by using a height-map sketching technique as guidance for a patch-based texture synthesis of a terrain. As opposed to the method by Gain et al. [48], they provide low and indirect control over the height and boundary of the resulting landform, when they choose the type of model as the input example. To reach an intuitive 3D sketch, curves are projected on an existing surface, as opposed to defining strokes from different view-points. None of these methods target the generation of subsurface illustrations as we do.

Following the classification given by Joshi [61], our technique falls in a three dimensional shape modelling category named *curve-based modelling*. Amongst the three further subdivisions of the category, we fit in *extruding 2D shapes* (the other two are: *inflating 2D shapes* and *drawing 3D curves*).

A.4 Methodology

Our aim is a rapid sketching tool for creating illustrative visualizations of structural geology, as seen in geology text books and as used in exploration companies to describe subsurface situations. Structural geology is the study of the three-dimensional distribution of rock units with respect to their deformational histories.

Models in structural geology basically consist of stacked layers in a so called layer-cake configuration. Therefore, we construct our model layer by layer. Each layer is represented by its boundary surface, defined by a curve on a 2D cross section. The user simply draws the top and the bottom boundaries of the layer (see for instance the upper part of Figure A.2). It is possible to add as many layers as desired. At any time, strokes can be selected and redrawn. Boundary strokes can be folded by selecting one or several of them and defining a deformational stroke. The deformational stroke will vertically offset the selected layer boundaries (see red stroke in Figure A.6). After the folded layers are defined, the user can fault them by sketching a fault curve and defining a force (see Figure A.7). Each layer can be textured with individual patterns that represent materials such as shale, silt, sand or salt. The textures are automatically deformed according to the folded shape of the layers and are discontinuous over faults. However, the texture deformation can be overridden by the user to represent a particular orientation, deformation or erosional history of a layer. To reach the final visualization in a quick fashion, the user's strokes are acquired on a vertical planar slice (technically called a *"geological section"*) and then extruded into three-dimensional space. After extrusion, the user can draw on the top and side surfaces of the 3D model to add details.

A.4.1 Folding

The starting point for creating a sketch, is a blank window representing a geological cross-section. Here, every stroke defines a boundary of a layer. Each stroke is stored as a sequence of line segments. When defining a fold, any kind of stroke that does not self-intersect is allowed.

The user can select one or several boundaries and apply folding (see Figure A.6). Folding is achieved by defining a new stroke that deforms the selected boundaries; similar to the approach by Bujans [20]. This new stroke does not have to be of the same length as the boundaries. In the length interval of the folding stroke, the y values of the selected boundaries are displaced according to the y values of the stroke.

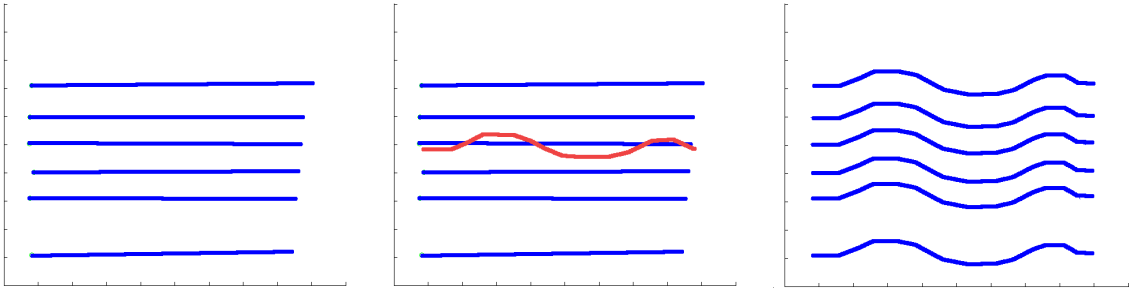


Figure A.6: Fast global folding of horizontal layers by defining a deformational stroke in red that deforms all layer boundaries equally.

Technically, user's strokes are acquired as piece-wise linear approximations of the samples along the stroke. Each planar region enclosed by two consecutive boundaries is

triangulated. For this purpose we use a constrained Delaunay triangulation (employing CGAL [2]) to deal with non-convex polygons, where the polygon is defined by the two boundaries of the layer and constraints are given by the edges of the polygon. This constitutes the initial mesh. The mesh will need to be refined when we will later use a conformal map to generate the texture for the layer.

If we do not need to add faults in the model, we can extrude the drawn layers and get the first approximation of the structure we have in mind to represent. The fold geometry, represented in an xy plane, is extruded in the z direction. The curves represent surface interfaces of different geological bodies and serve as an initial texture parameterization discussed in Section A.4.3. After that, the layer is ready to be visualized. Default colours are automatically applied to each layer when the model is ready.

As an alternative to sketching abstract ideas, the user can trace out folds and fault structures on a background image, trying to represent the real situation observed on the study field.

A.4.2 Faulting

To create a fault, the user sketches a curve on the geological section and specifies the direction of forces acting on the fault (referred as "*displacement direction*" in the figures), on either side of the fault. As in nature, if the stress is fault convergent, i.e. it pushes the sides divided by the fault together, the overlying block of the fault (the *hanging wall*, see Figure A.7), moves over (*reverse fault*, as in Figure A.4) the underlying block of the fault (the *foot wall*, see Figure A.7). On the other hand, if the stress is fault divergent, i.e. it pushes the sides apart, the *hanging wall* shifts down (*normal fault*, as in Figure A.7 and the example in Figure A.8) along the curve representing the fault (see Figure A.7). Geometrically, the fault divides each intersected layer in two new layers. Two types of faults are feasible: the so called *thrust fault* (see Figure A.7), where the fracture also intersects the top surface, and the *blind thrust fault* (see Figure A.4), where the fracture does not reach the top surface.

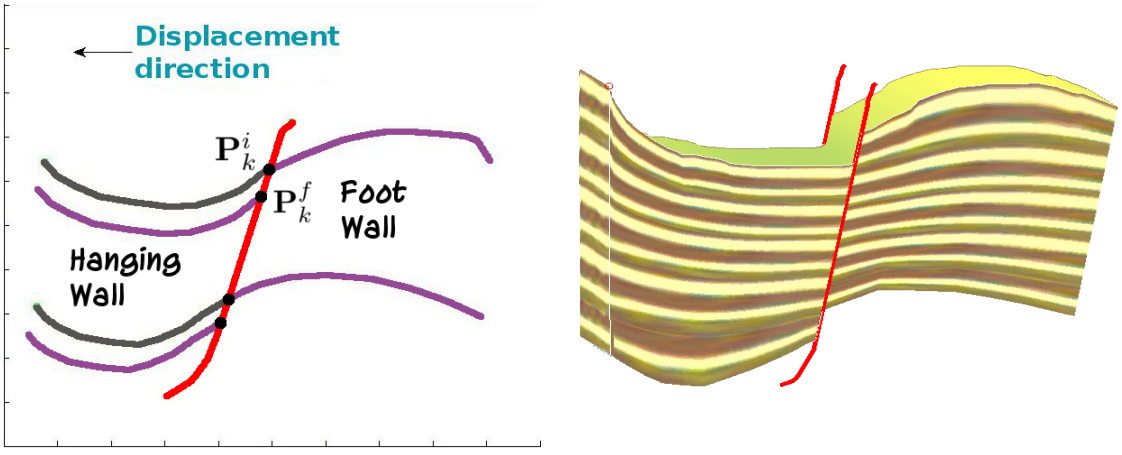


Figure A.7: Example of a textured layer which has undergone folding and faulting.

The system derives from the displacement direction whether the force is pointing towards the fault or away from it. From the trace of the fault, we detect which boundaries are involved and their corresponding intersections. The displacement of layers generated by the fault is proportional to the length of the sketched vector. Depending on the

inclination of the fault, the direction of the vector and which side it acts on, we move the point of intersection of each boundary with the fault by shifting it up or down along the fault. Subsequently, we apply the same translation to the entire boundary. Boundaries on the side of the foot wall remain in their position, while boundaries on the side of the hanging wall move of a vector $\mathbf{s}_k$. The displacement vector $\mathbf{s}_k$, related to the k-th boundary, is defined by $\mathbf{s}_k := \mathbf{P}_k^f - \mathbf{P}_k^i$, where $\mathbf{P}_k^i$ is the point of intersection of the k-th boundary with the fault before faulting, whilst $\mathbf{P}_k^f$ is the final position of the point $\mathbf{P}_k^i$ after being moved along the fault trace. The point moves along the fault until the length of its path is equal to the required displacement.

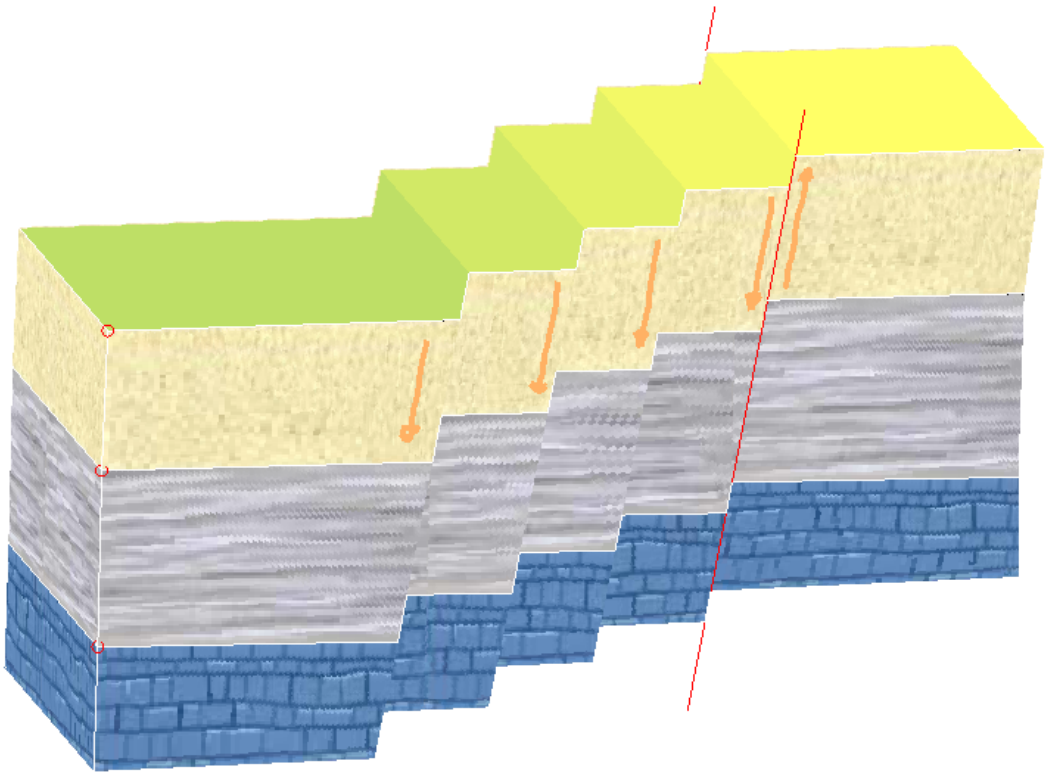


Figure A.8: An example of multiple faults.

When applying a fault on a layer, two cases may occur: either the layer remains connected or the displacement due to the fault completely separates the entire layer in two blocks. In the latter case, we must change the global structure of the model by splitting the layer, originally defined by two boundaries, into four boundaries and keep track of which boundaries belong to the same layer block. Consequently, we have to separate the surface that was representing the unfaulted layer.

A.4.3 Texturing

To let each layer represent a type of rock material, we allow the user to place a texture on its visible sides. We let the texture follow the shape of the sketched layer for giving an idea on how the material compresses or deforms under the action of the physical forces (see Figure A.9 for our texture application and deformation). The deformation is initially defined by the boundaries of each layer (Figure A.9, top), however it can be overridden by the user according to a new sketch (Figure A.9, bottom).

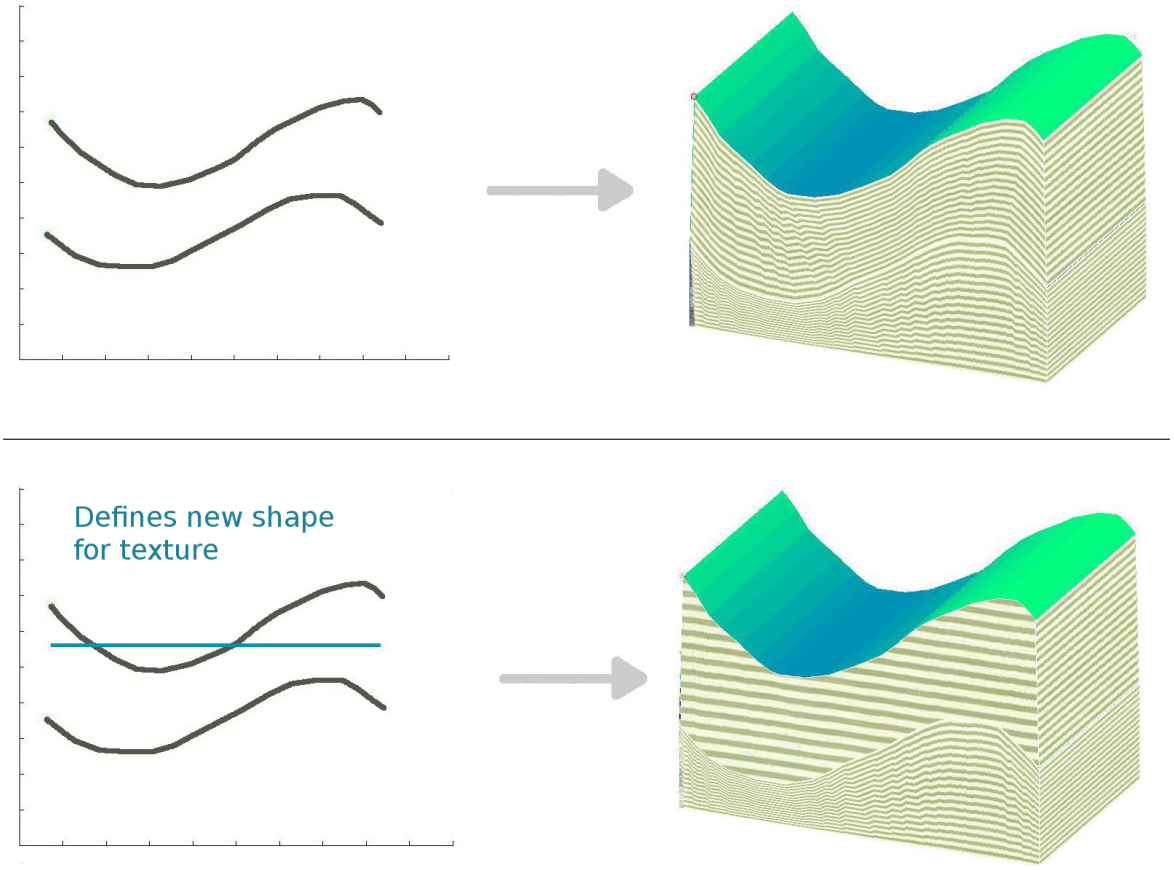


Figure A.9: Layer texturing: default texture shown in the top image, while a new sketch (bottom-left image) defines the modified shape of the texture in the upper layer (bottom-right image).

The textures used in geology are symbolic and communicate different rock types. Figure A.10 shows a legend of different rock symbols. When shearing such textures, their symbols might become unrecognizable. To maintain the repeating patterns in textures as recognisable as possible after deformation, we perform an angle preserving (*conformal*) parameterization. A conformal map lets us achieve a more recognizable, robust and aesthetically pleasing result.

Conformal maps preserve angles when mapping from model space to texture space. A conformal map allows the texture to faithfully follow the behaviour of the boundaries and minimizes distortion that may occur during texturing. Prior to the conformal mapping we refine the mesh using one-to-four triangles up-sampling and then we employ a conformal parametrization.

As described by Floater and Hormann [44] and by Mullen et al. [86], to get a conformal map $\mathbf{u} : \mathbb{R}^3 \to \mathbb{R}^2$ (see Figure A.11), we have to minimize the *Dirichlet energy*

$$E_D(\mathbf{u}) = \frac{1}{2} \int_{\chi} |\nabla \mathbf{u}|^2 dA,$$

where χ is a differential surface patch and $\nabla \mathbf{u}$ indicates the gradient of the function $\mathbf{u}$. Let $\mathcal{A}(\mathbf{u})$ be the area of the image of $\mathbf{u}$. Since we know that $E_D(\mathbf{u}) \geq \mathcal{A}(\mathbf{u})$ [108], we get a conformal map by imposing the energy $E_C(\mathbf{u}) := E_D(\mathbf{u}) - \mathcal{A}(\mathbf{u})$ to be zero. To calculate this in a discrete environment, which in our case is a triangular mesh, it is

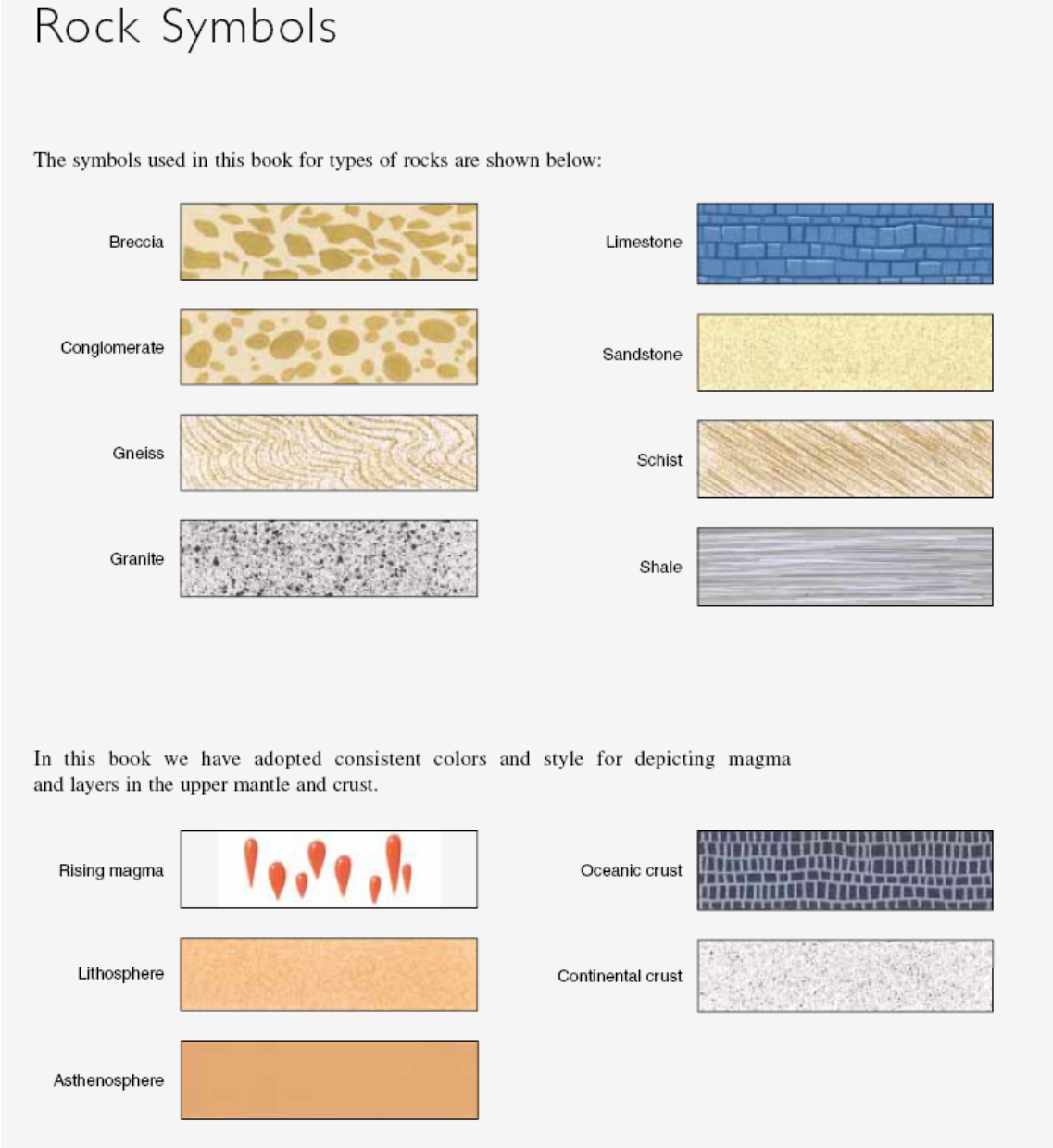


Figure A.10: Example from a geological book [131] of a legend of different rock symbols, courtesy of Graham R. Thompson.

necessary to discretize the map $\mathbf{u}$. This has been solved using the *Discrete Conformal Map (DCM)* [38] or the *Least Squares Conformal Map (LSCM)* [70]. For *LSCM*, the aim is to have the gradient of the u coordinate and the gradient of the v coordinate orthogonal and with the same norm. But it is known [86] that *LSCM* is equivalent to minimizing $E_C(\mathbf{u})$ and we use it in our implementation. *LSCM* lets us achieve good behaviours of the texture following the shape of the two boundaries (Figure A.1).

We give the conformal map algorithm these constraints:

- a strip which represents a layer in the model is a closed surface and needs to be opened with a proper cut, for instance along a vertical edge of one of the sides;
- *pinned* vertices [70] are chosen to be the vertices on the boundary of the cut strip. They are mapped to the boundary of the texture accordingly to the piece-wise

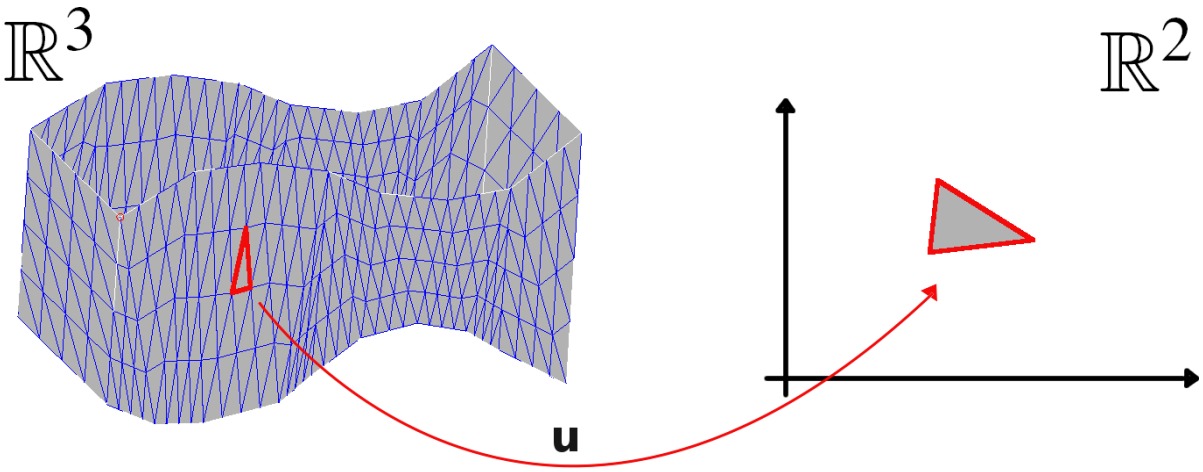


Figure A.11: Conformal map **u**.

linear distance along the boundary, from an arbitrary fixed vertex. They do not change during the *LSCM*, but they are used for computing internal vertices;

- internal vertices change their position during *LSCM*, but we maintain their connectivity which we have previously obtained applying one-to-four triangles upsampling.

Then the conformal map is calculated which satisfies these constraints and guarantees an aesthetic and angle-preserving texture in between.

In presence of faults, the texture image has to be split according to their number and position within the layer. Afterwards *LSCM* is applied to each separate part independently.

Figure A.12: New texture, obtained on the basis of a user's stroke (split in $\mathbf{s}_1$ and $\mathbf{s}_2$). $\mathbf{b}_1$ and $\mathbf{b}_2$ are the top and bottom boundary of the layer and they initially deform the texture, while $\mathbf{s}_1$ and $\mathbf{s}_2$ are copies of the user's stroke that is acquired to guide the behaviour of the texture.

The user has the option to override the default texture constraints, initially taken from the sketched boundaries, as done by Zhang et al. [145] for solid texturing. This

is useful for describing different types of erosional situations. For instance, a layer that once was horizontal might have lost material due to a glacier sliding on top of it. This results in the top boundary becoming concave. In this case we want the texture to keep its horizontal shape to communicate the depositional history, and not be affected by the top boundary. Another example is when the user wants to represent a situation where a fold arises from a river erosion as opposed to ground compression. In these cases, the user selects the layer and changes the texturing with a simple sketch (Figure A.9 shows how this effect can be reached in a few steps). The new input stroke $\mathbf{s}_1$ is duplicated and set from its copy $\mathbf{s}_2$ on a distance which let the whole layer fit in between. Therefore, if the layer is identified by its two boundary curves $\mathbf{b}_1$ and $\mathbf{b}_2$, then

$$\min\{\mathbf{s}_1\}_y \geq \max\{\mathbf{b}_1\}_y \quad \text{and} \quad \min\{\mathbf{b}_2\}_y \geq \max\{\mathbf{s}_2\}_y.$$

Afterwards, the texture deforms according to the new pair of strokes ($\mathbf{s}_1$, $\mathbf{s}_2$) and the image for the layer is extracted from the inside (see Figure A.12). Alternatively, the user can specify two boundaries instead of one. This additionally allows to express compression, as shown in Figure A.13.

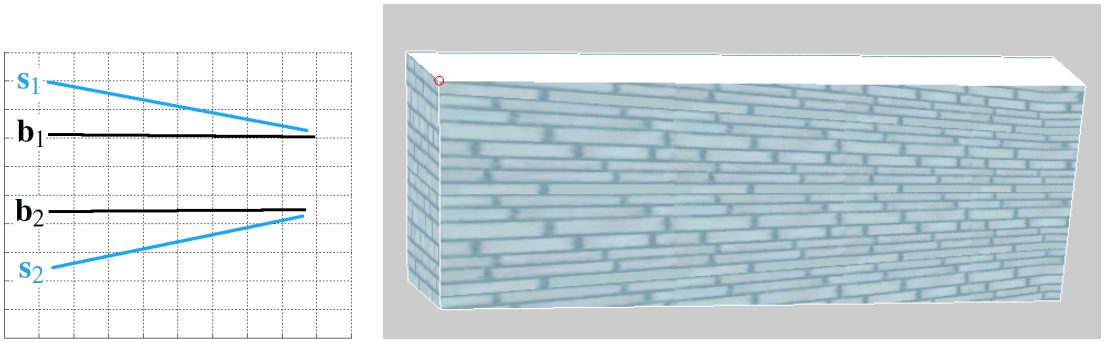


Figure A.13: Changing the default texturing, that would follow the layer boundaries $\mathbf{b}_1$ and $\mathbf{b}_2$, with two new sketches, $\mathbf{s}_1$ and $\mathbf{s}_2$, to convey layer compression.

A.4.4 Projected Drawing

Drawings such as those shown in Figure A.5 and arrows in Figure A.8 are obtained by sketching curves or patches directly on the 3D model. A ray that starts from the viewpoint intersects triangles while the sketch is drawn and the curve is directly projected on the model. The user can choose colours and thickness of the curves. Projected drawing makes it possible to define, for instance, rivers, lakes, magma intrusions, but also explanatory arrows, handmade notes or labels for features of the model.

We perform automatic colouring of the terrain based on its height values.

A.5 Results

As mentioned earlier, we aim to obtain, from scratch, a layer-cake visualization that is in correspondence to the illustrations used in the text books for explaining geological events in the crust of the earth.

All the models that appear in the figures of this paper have been constructed with very few curves, nevertheless these few input strokes let us represent and express many

different situations. In short time (less than a second on an *Intel Xeon E5620 CPU*), the system generates the geometric structure of the layer-cake model. To compute the conformal parameterization, one or two seconds more are needed. The amount of time depends on the number of samples which have been used to acquire and store the boundary curves defining all the layers.

The implementation has been done in Matlab. The code is not optimized and we do not make use of GPU acceleration. For this reason, we believe that the computational time can be further reduced. In any case, with our implementation, a user that wants to represent a model can easily do it in less than a minute, starting from an empty canvas to the final 3D illustrative visualization. This is less than the time a person would employ to draw the same concept on a physical paper with a pencil or on a PC with a painting software (as shown in Table A.1, which compares required times using our method and a more classical approach). Furthermore, in both cases, one ends up with a 2D picture instead of a 3D model. These are some of the goals that a sketch-based technique aims to achieve. Table A.1 shows an approximate comparison of times that are necessary to create illustrations such as in some of the figures in our paper. The second column contains the required time for a user to sketch curves and choose textures. The third column lists the processing time including time for performing the conformal map. The last column shows the time that would be necessary to create a similar model of the corresponding figure, according to an estimate given by the same geologist and illustrator who produced Figure A.3.

Time	User interaction	Processing	Illustrator's estimation
Figure A.1	$\sim$ 20 sec.	$\sim$ 2 min.	$\sim$ full day
Figure A.8	$\sim$ 20 sec.	$\sim$ 1 min.	$\sim$ 1-2 hours
Figure A.14	$\sim$ 20 sec.	$\sim$ 1 min.	$\sim$ 1 hour
Figure A.15	$\sim$ 20 sec.	$\sim$ 1 min.	$<$ 1 hour

Table A.1: Approximate comparison of times.

Our method allows for creating either illustrative-style (Figure A.8, A.14) or photo-realistic style (Figure A.1) images depending on what class of textures is used. The former is best suited for discussing and brainstorming of scenarios during subsurface exploration and for creating illustrations for pedagogical and presentational use. The latter can be used for describing different outcrop constellations to geology students and for relating to observed field data.

Regarding the use of the default texture application (adapting to the limit boundaries) with respect to the texture that is deformed by a user's sketch, we have examples of the first eventuality in figures A.1, A.7 and A.14. We observe examples of the second case in figures A.2 (bottom-right image), A.5 (right image, middle layer) and A.9, where the layer on which the texture has been re-adapted, following the new shape defined with the input stroke, is easily recognizable. Figure A.15 shows how to give to the illustrative model the appearance of being submerged (left image), by simply adding transparency to surfaces of the top layer, and how to better convey a context by adding a few projected drawings on layers (right image).

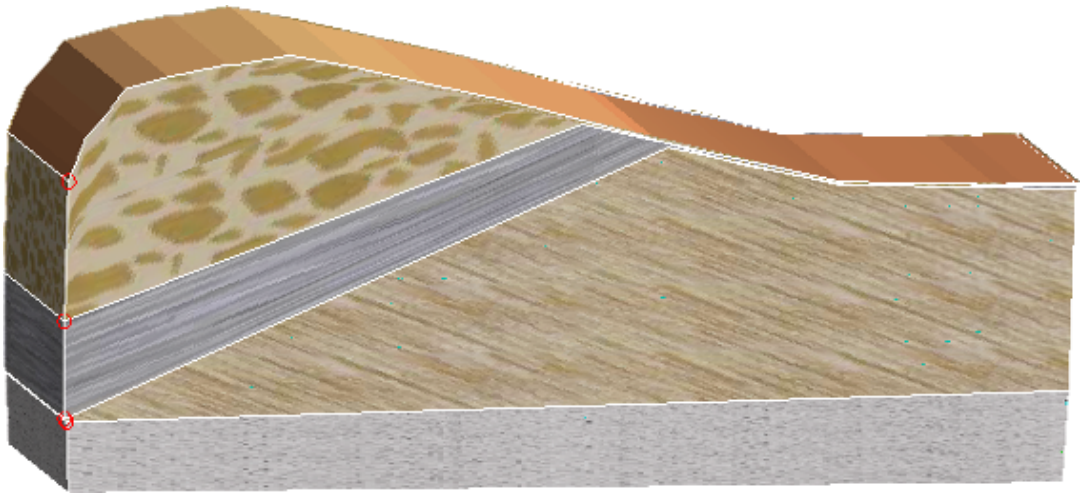


Figure A.14: Example with illustrative textures.

A.6 Conclusions

We have developed a tool for sketch-based modelling, which is specific for the construction of stratigraphic structures of the subsurface with few strokes. The whole process leads to an illustrative 3D visualization (layer-cake representation). We have avoided procedural modelling in favour of getting more control over the generation of the model. All has been done following guidance from professional geologists and illustrators. They found our tool helpful for them in many ways: it solves needs described in Section A.2 (e.g. simple, rapid, illustrative, interactive) and can be a supplement to their current approach to generate illustrations. This is because of interactivity of 3D models and handle-ability of textures on layers.

Such a tool can be a new way for users, in particular geoscientists, who want to share visual thoughts, that originates as abstract ideas, and communicate them through a digital representation.

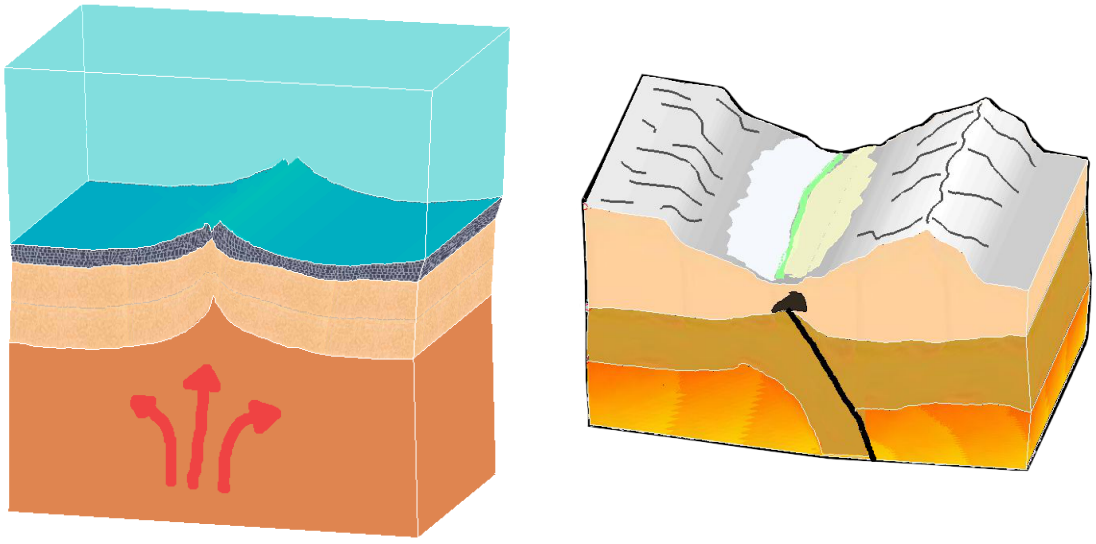


Figure A.15: Example of the oceanic crust with red arrows suggesting movements of the mantle (left) and model with projected drawings that enhance context (right).

A.7 Future Work

An interesting way to face the same problem of generating 3D illustrative stratigraphic representations is to interpret the process going backwards in time. That is, concerning faults, instead of defining the crack and then the displacement of the layers, we start sketching an already existing fault process (perhaps coming from an observation on the field) and we reassemble/put in correspondence blocks of the same rock that belong to the same original layer.

Acknowledgment

The authors would like to thank all reviewers that gave comments and suggestions for improvements, Haakon Fossen for authorization to Figure A.3, valuable feedback and expertise and Graham R. Thompson for authorization to Figure A.10. This work is funded by the Petromaks program of the Norwegian Research Council through the Geoillustrator project (#200512).

Paper B

Sketch-Based Modelling and Visualization of Geological Deposition

Mattia Natali[1], Tore Grane Klausen[1], Daniel Patel[2,1]

[1]University of Bergen, Norway
[2]Christian Michelsen Research, Bergen, Norway

Abstract

We propose a method for sketching and visualizing geological models by sequentially defining stratigraphic layers, where each layer represents a unique erosion or deposition event. Evolution of rivers and deltas is important for geologists when interpreting the stratigraphy of the subsurface, in particular for hydrocarbon exploration. We illustratively visualize mountains, basins, lakes, rivers and deltas, and how they change the morphology of a terrain during their evolution. We present a compact representation of the model and a novel rendering algorithm that allows us to obtain an interactive and illustrative layer-cake visualization. A user study has been performed to evaluate our method.

This article was published in *Computers and Geosciences 67C*, 2014, pp. 40–48, DOI: http://dx.doi.org/10.1016/j.cageo.2014.02.010.

B.1 Introduction

Geologists are interested in better tools for externalizing their ideas on the earth's behaviour. They want to do this in an expressive and simple way, which is particularly important for communicative purposes. Current modelling tools in geology have a high learning curve and are tedious to use (Caumon et al. [23]). We present a simple sketching interface and a data structure for compact representation and flexible rendering of subsurface layer structures. This is useful for representing mountains, basins, lakes, rivers and deltas. Rivers and deltas change the morphology of a terrain through erosional and depositional processes. This is left as imprints in layers of the terrain. Our technique provides a way to sketch and visualize such layers (Figure B.1). Our representation is well suited to obtain interactive layer-cake visualizations, resulting in geological illustrations that are helpful for education and communication. Conventional hydrocarbon reservoirs (and aquifers) are found in porous bodies of rock. Examples of such rock bodies include sandstone, which is found in sedimentary basins and has a high preservation potential (Hinderer [56]). The sandstone might, under favourable circumstances and the right basin development (where hydrocarbon source rock and reservoir seal is present), become a reservoir for hydrocarbons. This is, in a crude sense, the reason why these rock bodies receive a lot of focus in geology, and a motivation to try to understand them to the full extent that data allows.

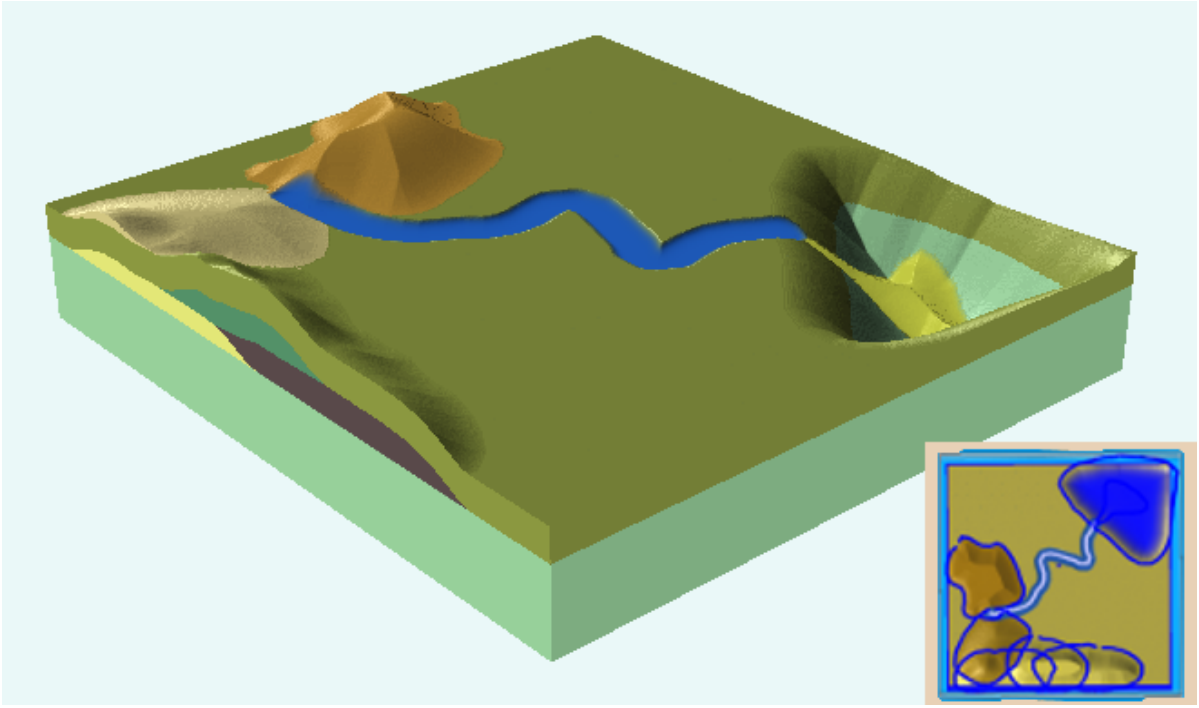


Figure B.1: A 3D model created using our sketch-based approach to shape surface and subsurface geological features. Bottom right inset shows a map view of the sketched strokes in blue overlaid on the model.

Available data (e.g. seismic, well logs or core) can often be of limited resolution and extent. In these cases, geologists have to develop conceptual ideas to describe the shape of rock bodies, and, from that, which processes were involved in their deposition. The processes involved in the deposition of sandstone bodies are known to vary between different depositional environments, and can thus be differentiated based on observations

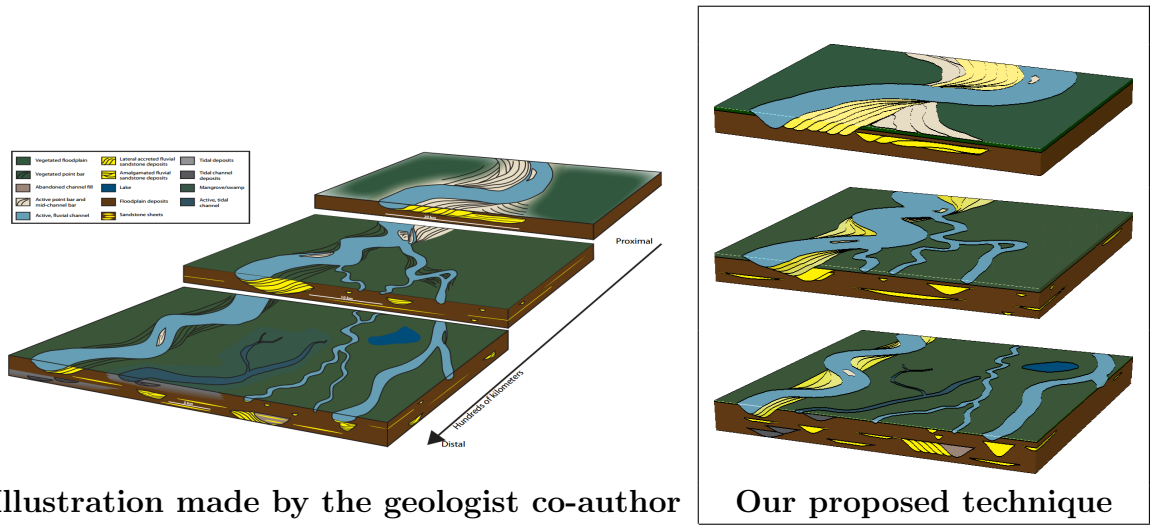


Figure B.2: Comparison between a 2D hand made illustration of a river sedimentation process (left) which took one working day to produce and the result achieved with our approach (right) which took one hour to produce. The example is a real case analysis built from field observations.

and interpretation (Reading [117]). To develop good conceptual models for the reservoir sandstone, their horizontal and cross-sectional characteristics are often highlighted by schematic block diagrams (e.g. Gani and Bhattacharya [50], Porębski and Steel [110]). In Figure B.2 left, an example of a hand-drawn block diagram depicting a meandering river channel is shown. It illustrates the aerial and the cross sectional expression of sandstone point bars (sediments deposited along the inner bank of a meandering stream) and how one sandstone body is overlaid by another. Internal architecture is important in the sense that it tells how the depositional element (e.g. channel or delta) evolved, and subsequently how small-scale heterogeneities such as mudstone might be distributed within the overall sandy body. This can have direct implication for hydrocarbon fluid extraction. Our approach offers a new way of producing illustrations by performing interactive erosion and deposition that lets the illustrator mimic processes that she interprets to have been the cause for the sandstone deposition. An additional consequence of our 3D sketched models is their manoeuvrable cutting planes that enable multiple cross-section visualizations. This helps in understanding complex internal layering within the sandstone, otherwise not intuitively apprehensible (Bridge [17]). Among the depositional systems that may result in hydrocarbon accumulation, rivers and deltas are central.

In summary, our overall contribution is a sketch-based system that makes it possible to quickly build 3D interactive geological illustrations from scratch. Sequentially defining alterations on a model is less laborious than drawing 2D illustrations; this opens up for discussions, fast hypothesis testing and creation of time-series illustrations. A novel central aspect lies in the proposed data structure (each stratigraphic layer corresponds to a composition of one or more heightmaps) and the way it is processed to render volumetric models. The main features that characterize our tool consist of operators that interpret each sketch to generate deposition and erosion processes; in particular, rivers and channels, mountains, basins, deltas and intermediate stages of their evolution.

B.2 Related Work

Little work exists on modelling and visualizing deltas in computer graphics. For modelling rivers and erosion in a geological setting, most of the methods are based on fractal noise generation and physical based processes (for instance the works of Benes et al. [13] and Stava et al. [127]). Such procedural approaches reduce the degree of control over the landscape development. Other sketch-based techniques have been introduced to model landforms on a terrain ([57, 48]), but they only consider the top surface and not subsurface structures, as we do. Some work in geology focuses on producing layered representations of the earth by modelling the mesh of each separating surface individually, like Baojun et al. [8], but are not sketch-based and therefore time consuming. Amorim et al. [5] address 3D seismic interpretation by modelling one surface at a time, while Cutler et al. [34] propose a procedural method to obtain layered solid models. None of these two last works consider fluvial systems consisting of lakes, rivers or deltas.

A state of the art report on modelling of terrains and subsurface geology was presented by Natali et al. [90]. There two works are reviewed that describe volumetric representations applicable in geology, although they do not discuss how to sketch them fast as we do. Takayama et al. [130] obtain layered models using diffusion surfaces. Wang et al. [140] represent objects using signed distance functions. Composite objects are created by combining implicit functions in a tree structure which makes it possible to produce volumes made of many smaller inner components.

Geostatistics Surface- and event-based modelling have recently been developed as a branch of geostatistics. Xie et al. [144] suggest depositional and erosional surface-based models as collections of separate sediment units, each of them stochastically generated. Lopez et al. [74] create their 3D models, in the context of meandering channelized reservoirs, combining a process-based approach to model river motion and deposition, and a stochastic approach to respect vertical proportion curves that reflect natural behaviour. Pyrcz et al. ([115] and [114]), Michaels et al. [83], and Abrahamsen et al. [4] use statistical analysis on process-based models to create surface-based distributary lobe models with the addition of available geologic information. Pyrcz et al. [116] present new applications for event-based models and methods to generate them with hand editing. Bertoncello et al. [15] constrain sedimentary surface-based models to field data (wells and thickness data interpreted from geophysics, such as seismic data).

Geological Illustrations Geological illustrations let geologists express and explain subsurface processes of how the earth has evolved and can be used to predict and discuss future scenarios. In addition to enabling communication between experts, geological illustrations are heavily employed for teaching purposes as they are expressive and easy to comprehend. This process of externalizing ideas is a crucial aspect in geology (as highlighted by Lidal et al. [73]).The added value of our 3D models with internal structures compared to static 2D illustrations made on paper or with 2D raster or vector drawing software is the support of interactive exploration. Creating a volumetric 3D model allows for visualizing arbitrary cross-sections of the model and ensures spatial consistency of internal structures. For hand-drawn 2D illustrations, spatial consistency between cross-sections is not ensured as features can be drawn independently on each cross-section without honouring the underlying 3D structures. We are not aware of any work similar to ours for fast sketching of 3D geological illustrations containing internal

structures. Related is the work by Natali et al. [91] which is able to imitate 3D geological illustrations by supporting 2D drawing on the surfaces of a hollow 3D boxlike object.

Fluvial Landscape Evolution When the top surface of a geological model contains rivers and streams, or has been shaped by them, it is called a fluvial landscape. A clear classification of different methods that have been used to model or simulate a fluvial system and the surrounding landscape can be found in the work by Coulthard and Wan De Wiel [32]. Models concerning fluvial geomorphology are made that attempt to retrieve the history or to predict the evolution of rivers (Figure B.2 is an example of a fluvial geomorphology illustration). Such models are also necessary when a large temporal scale process is considered and cannot be studied with just observations of the real world, which are in a small temporal scale compared to the river history. The work on river geomorphology concerned with simulation of erosion and sediment transport usually utilizes physically-based constraints such as the conservation of mass and momentum of the river flow, formally expressible with the Navier–Stokes formula.

Layered Representations A layered heightmap representation has been proposed earlier ([11]), but only in the context of terrain visualization and erosion, and not for representing subsurface structures as we propose. A representation related to ours is used by Lemon and Jones [69], employing meshes instead of heightmaps. They present an approach for generating solid models from borehole data. Their model construction is simplified by representing horizons as triangulated surfaces, where vertices maintain the same (x, y) position for every surface, and only the z-value varies. This simplifies intersection testing and topological relations between surfaces. Inspired by this work, we use heightmaps to represent layers and replace intersection tests with simple and fast arithmetic operations. In addition, correct topology between layers is implicitly maintained. Invalid layers, layers with holes, or empty spaces inside models are automatically avoided. Peytavie et al. [107] also use a layered representation, where subsurface layers are discussed, but only for the purpose of realistic modelling of the top terrain surface. The layers represent materials that interact at the top surface such as air, water, boulders, sand and ground. We model the top surface, but we also include internal structures and landscape evolution (Figure B.2 and Figure B.3) which can be rendered volumetrically and with arbitrary cuts.

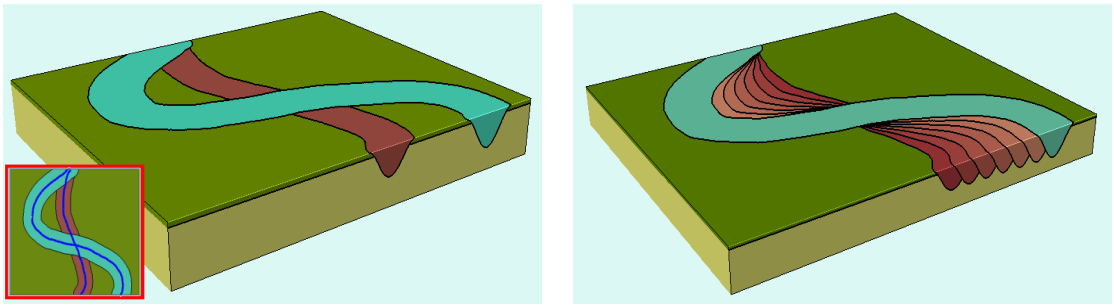


Figure B.3: River evolution example. Left: first and last configuration of the river are sketched and imprinted. Right: imprint of additional five intermediate stages of the depositional history.

B.3 Description of our Approach

The entire approach is based on two synchronized data structures; the *relative* layers and the *absolute* layers. The relative layers let us keep geological processes independent of each other with respect to time, allowing us to rearrange layers in any order without further computations. The absolute layers is derived from relative layers and is used for fast rendering. The i-th *absolute layer* $h_i^{abs}(x, y)$ is defined in each of its points $(x, y) \in \mathcal{G}$ by the height of the top surface of the layer in the reference coordinate system of the model, $\forall i = 1, \ldots, n$. Where n is the number of layers, $\mathcal{G}$ is a uniform grid and $h_0^{abs}(x, y)$ is defined to be zero in every point.

The i-th *relative layer* $h_i^{rel}(x, y)$ is defined in each of its points $(x, y) \in \mathcal{G}$ by the displacement

$$\Delta h_i^{abs}(x, y) = h_i^{abs}(x, y) - h_{i-1}^{abs}(x, y)$$

of two consecutive absolute layers, $\forall i = 1, \ldots, n$.

The relation between relative and absolute layers lets us express the i-th absolute layer as

$$h_i^{abs}(x, y) = \sum_{k=1}^{i} h_k^{rel}(x, y), \forall i = 1, \ldots, n.$$

Figure B.4 illustrates the relation between relative and absolute layers and shows how the final model is reached at each step i. The number of layers of the final composition is not known a priori, therefore we will only give a number to the last l layers, i.e. $L_i^1, \ldots, L_i^l$, with corresponding thickness values $T_i^1(x, y)$, ..., $T_i^l(x, y)$. In the third column of Figure B.4, for instance, we have: one layer each at steps 1 and 2, namely L_1^1 and L_2^1 , but with different shapes due to different thicknesses; two layers at steps 3 and 4, that is (L_3^1, L_3^2) and (L_4^1, L_4^2). Unique properties can be associated to each layer so that they can be represented differently in the final rendering.

B.3.1 Data Structure Description

A final composition of the model is made of a sequence of chronologically ordered relative layers (both positive and negative values are feasible). When, at step i and in a specific point $(\hat{x}, \hat{y}) \in \mathcal{G}$, $h_i^{rel}(\hat{x}, \hat{y}) \geq 0$, last layers of the final composition are not modified and a new one changes in $(\hat{x}, \hat{y})$ to have thickness $h_i^{rel}(\hat{x}, \hat{y})$. When $h_i^{rel}(\hat{x}, \hat{y}) < 0$, this affects previous layers and they must be changed accordingly. Figure B.5 gives us an example of this occurrence. The process develops as follows: the erosion expressed by $h_i^{rel}(\hat{x}, \hat{y})$ decreases the thickness $T_i^{l-1}(\hat{x}, \hat{y})$ of the previous layer; if $h_i^{rel}(\hat{x}, \hat{y}) > T_i^{l-1}(\hat{x}, \hat{y})$, then L_i^{l-2} is also involved and $T_i^{l-2}(\hat{x}, \hat{y})$ decreased; the iteration analogously continues until layer L_i^k, such that $h_i^{rel}(\hat{x}, \hat{y}) \leq \sum_{j=l-1}^{k} T_i^j(\hat{x}, \hat{y})$. In Figure B.5, L_i^2 and L_i^3, $i \in \{4, 5\}$, are set to zero thickness in our data structure and will not be visible in the rendered model.

Our model is obtained through an accumulation of relative layers, defined on uniformly sampled grids represented as 2D height maps. Each relative layer indicates the amount of deposition (positive values) or erosion (negative values) that generates a change in height with respect to the shape of the previous layer. For instance, to achieve a simple model made of a starting landscape and a river, the first relative layer (always also the first absolute layer, see top row Figure B.4) is used as an initial ground layer of constant thickness. Afterwards, the bed of the river is carved out by the second relative layer (second row in Figure B.4). This relative layer contains negative values

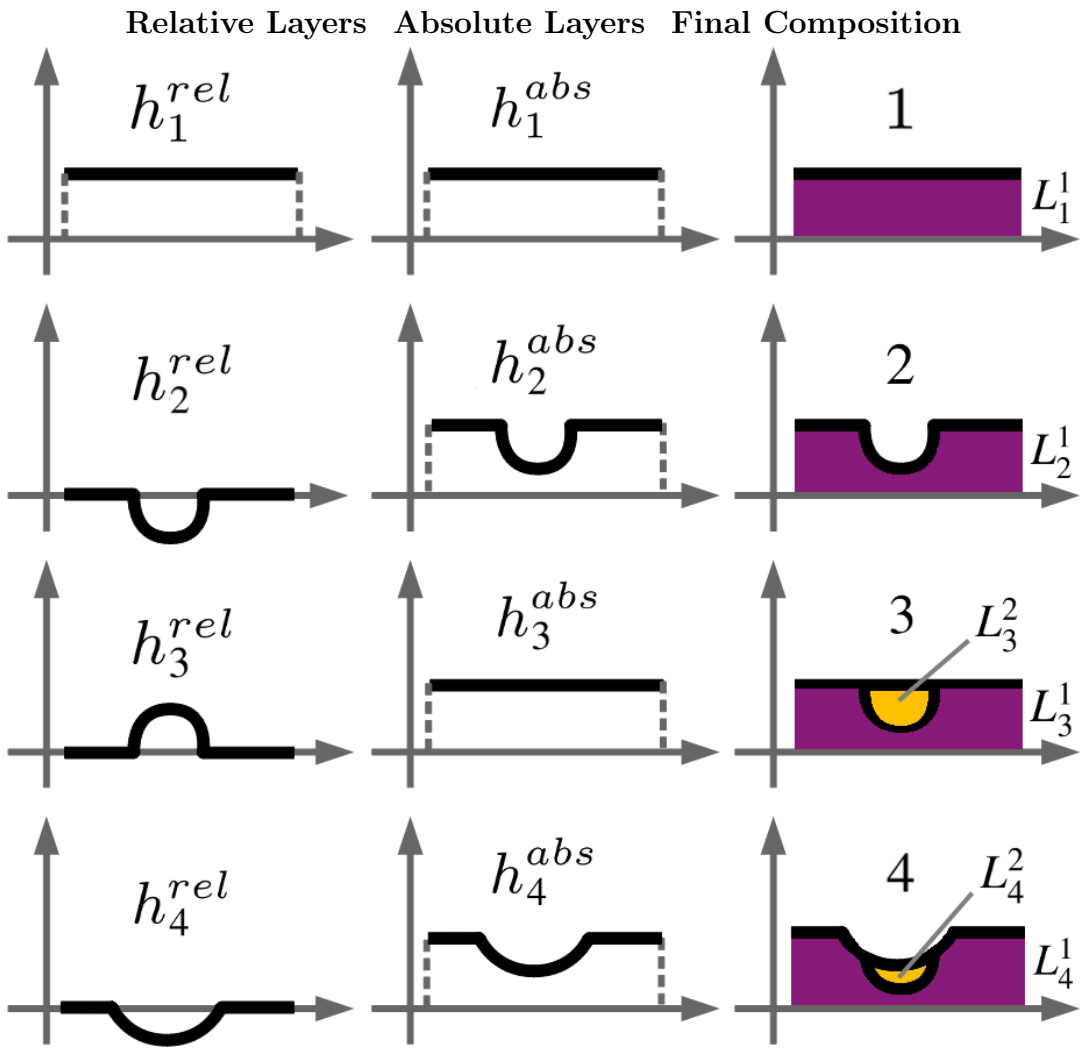


Figure B.4: Relation between relative layer, left column and absolute layer, middle column. The right column shows the accumulated final model.

that indicate the erosion depth for each point of the grid. Compared to using a boundary representation (Natali et al. [91]), we avoid time consuming triangle-intersection testing and vertex alignment between the boundaries of individual layers. The process of erosion and deposition is straightforward to realize and can be performed in parallel compared to constructive solid geometry algorithms performed on a boundary representation.

B.3.2 Rendering the Model

We visualize the model by applying ray-casting based volumetric rendering. Volumetric rendering inherently supports volumetric variations and translucency. Therefore our method can handle procedural 3D textures for defining different rock appearances, co-rendering of the model together with volumetric data such as seismic data of the area being modeled, or varying layer properties such as grain size in a delta deposit. As opposed to standard volumetric rendering which accesses a 3D array representing a discretization of the model space, we perform ray-casting directly on our compact absolute layer structure. The colour of each pixel on the screen is found by sending a ray from the pixel into the scene and accumulating the colour along the ray. For each sample position

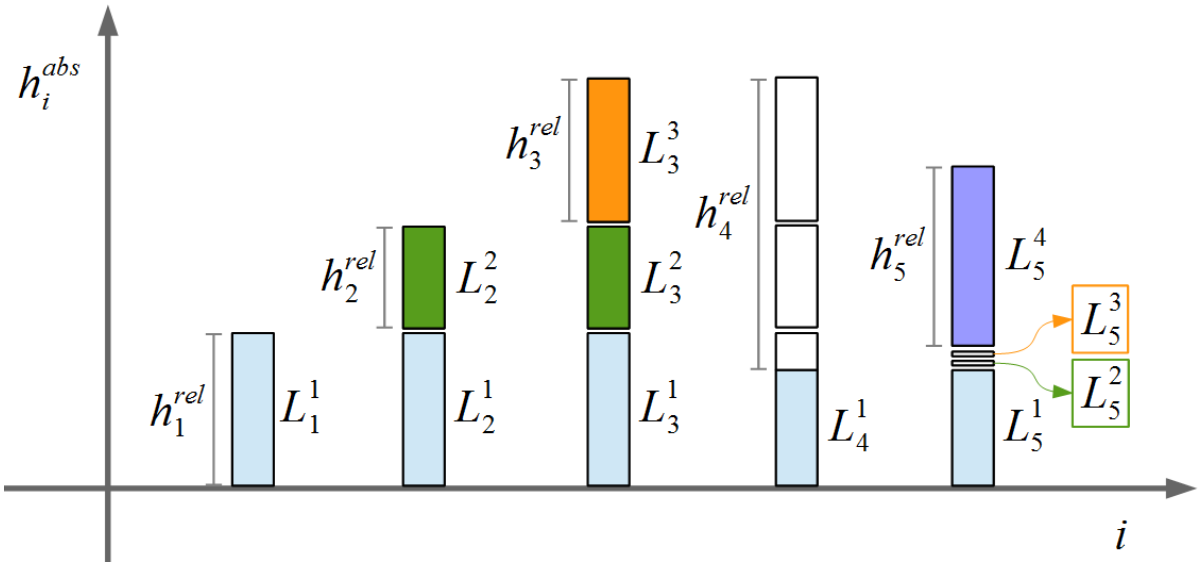


Figure B.5: Interaction between the i-th relative layer h_i^{rel} and the previous layers $L_i^1, \ldots, L_i^l$. This example refers to a single point $(\hat{x}, \hat{y})$ of the grid $\mathcal{G}$. Only $h_4^{rel}(\hat{x}, \hat{y}) < 0$, therefore from step 3 to step 4 we observe an erosion.

on the ray, we identify in which layer we are which decides which colour and opacity the sample has. We perform the ray casting on the GPU to enable fast parallel processing of rays.

Each layer can be considered as a 2D height field representing the thickness for each (x, y) position (as shown in Figure B.5). We store n layers, each with dimension $l \times m$, as a 3D texture with dimension $l \times m \times n$. For interactive rendering, we need a fast way to identify which layer a sample at position (x, y, z) is. Each (x, y) position in an absolute layer stores four components: the starting and the ending position of the layer and the index of the next and the previous nonempty layer. See Figure B.6 and the accompanying Table B.1 describing the four values for each of the layers for position $x = x_3$. In Figure B.6, the layer-index of the first sample at $x = x_1$, representing the first sample on each ray, is found by iterating over all layers until the interval containing the sample's z position is found. This search can be accelerated by performing a binary search. The search returns layer 2. For the next sample having $x = x_2$, we first look up in the same layer as the previous sample (layer 2), and retrieve the start and the end of the interval. The sample's z value is within this interval and we are therefore in the same layer. This lookup requires only one texture access. The next sample at $x = x_3$ has crossed a layer boundary. The content of all layers for position (x_3) is shown in Table B.1. The layer interval for L2 is below the z value for the sample, therefore we search in layers above to find the correct interval. We look up in the first nonempty interval L6 which the sample is inside.

With volume rendering we are able to change optical properties along the ray inside a layer as, for instance, a function of the distance to the source of deposition for communicating properties such as grain size (see Figure B.7).

On cutting planes, a black separator line is drawn between adjacent layers. This makes it easier to distinguish between different depositions. However it is possible to assign different layers into the same layer group to avoid separator lines between them. This is necessary for giving the brown soil in Figure B.2 one homogeneous appearance

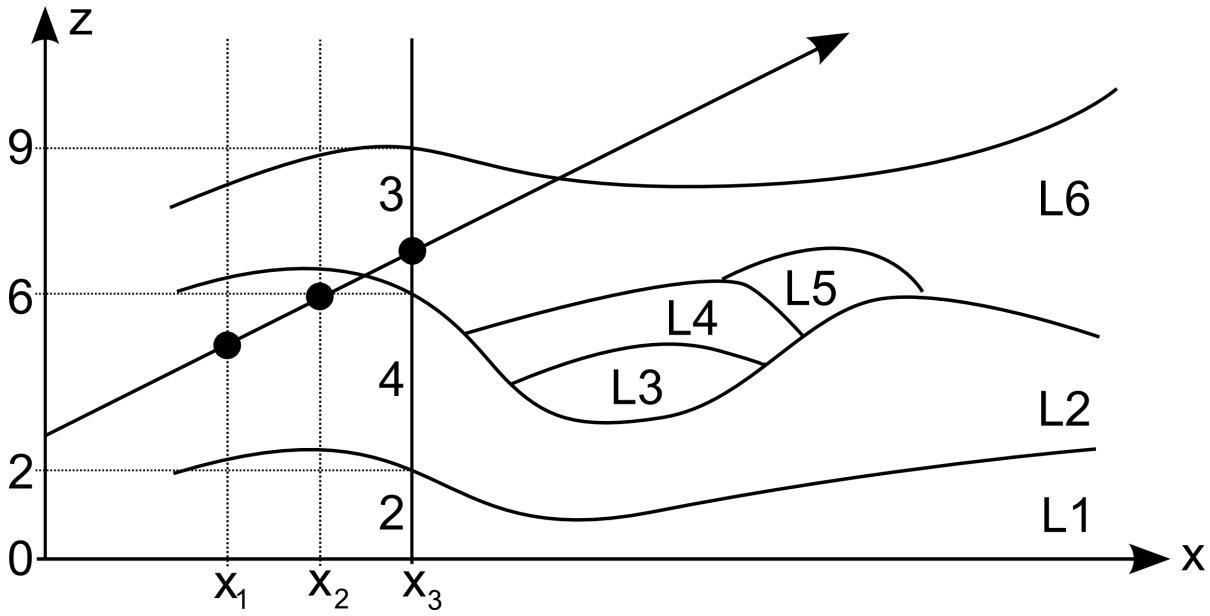


Figure B.6: Using the absolute data structure for fast access along a ray cast from left to right.

Layer	Start	End	Prev Layer	Next Layer
L1	0	2	-	L2
L2	2	6	L1	L6
L3	6	6	L2	L6
L4	6	6	L2	L6
L5	6	6	L2	L6
L6	6	9	L2	-

Table B.1: The absolute layer structure used for volume rendering. The table shows layer properties for position x_3 of the example in Figure B.6.

even though it consists of several smaller layers intermixed with river layers.

B.3.3 Geological Concepts Through Sketches

Sketching geological features on the model is based on two types of strokes: the open and the closed curve. They are the basis for all geological features. In addition, there is the possibility to add a layer of constant thickness for expressing a sedimentation process equally distributed in the landscape. Every sketched curve consists of an array of points that has been subject to a Chaikin subsampling ([25]) to reduce the number of samples while still maintaining its shape. In the following paragraphs, we describe how the sketches are interpreted, classified and converted to relative layers.

Modelling a River When the first and the last point of a curve are not close to each other (according to a predefined threshold), we consider the curve as a definition of the centreline of a river as seen from above (map-view). To allow branches in a river, the last drawn curve is merged with the previous one if they intersect.

Meandering rivers can be sketched by defining only the start and the end configuration

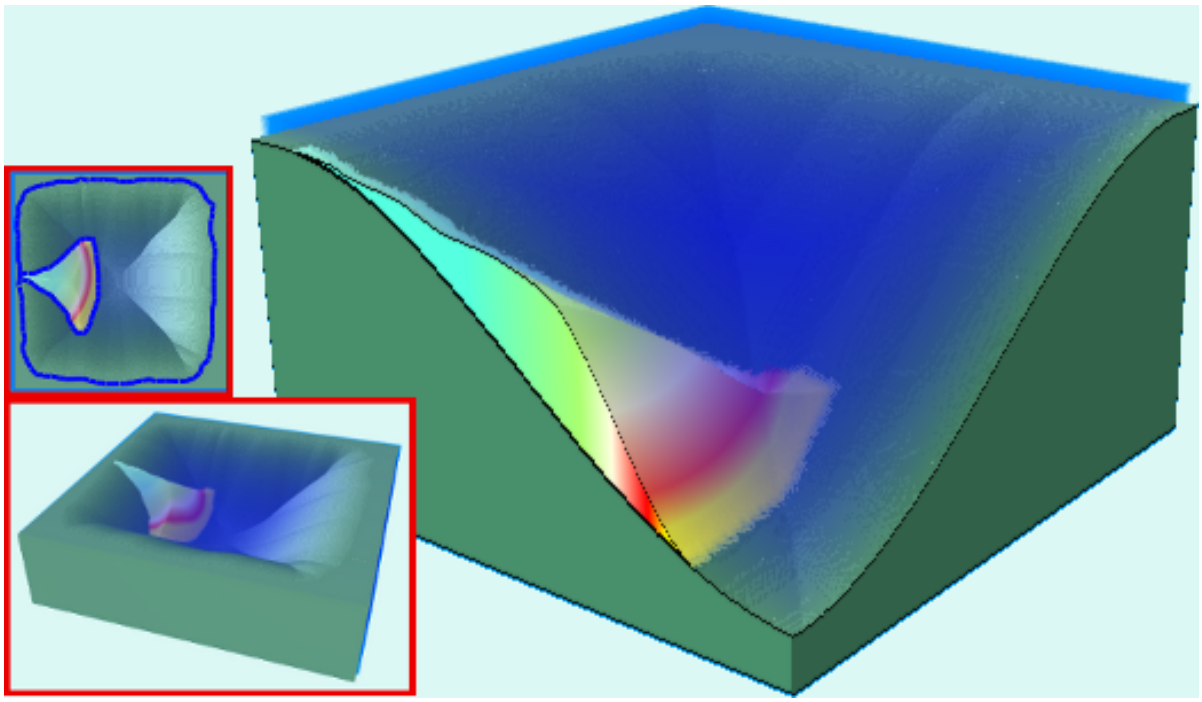


Figure B.7: Cross-sectional view showing the internal structure of a delta deposit, where different grain sizes are conveyed by colour transition. Water is set to be highly translucent.

and specifying the number of intermediate rivers. Intermediate rivers are then calculated as linear interpolations of the first and the last configuration. The interpolation between two curves is done by resampling both curves uniformly, so that they have the same number of points and at the same curve-length distance. Instead of a physically-based interpolation, we use an interpolation based on curve-length parameterization. The simplicity results in predictable behavior. This enables the user to express any kind of river evolution by manually sketching interpolating rivers. For simple cases, a start and end river is enough. For more complex cases, the user must sketch several key rivers which the system will then interpolate between.

Given two rivers, defined by two curves C_s and C_e, that respectively describe the starting and the ending configuration, we proceed as follow. Let's assume the behaviour of the river during its evolution is expressed by r intermediate steps and each curve is resampled into p points. Every intermediate curve C_i is defined as a set of vertices V_i^σ, where $\sigma = 1, \ldots, p$. The interpolation between a vertex V_s^σ on C_s and a vertex V_e^σ on C_e at intermediate step i is given by

$$V_i^\sigma = (1 - t)\, V_s^\sigma + t\, V_e^\sigma,$$

where $i = 1, \ldots, r$ and $t \in [0, 1]$.

We now introduce a mathematical description of our river sections. The flow of a river has decreasing velocity from the middle of its bed to its banks, and erosion is proportional to velocity. We model this with a function that smoothly goes towards 0 with the distance to the centreline (see Figure B.8 top), although our system can easily be extended to support any analytic or user-sketched definition of the river section. When the user draws the centreline of a river, a scaling factor for river depth and a scaling factor for river width can be set. Each eroding layer associated with rivers is always coupled

with complementary depositional layers, i.e. defined by grids with opposite values.

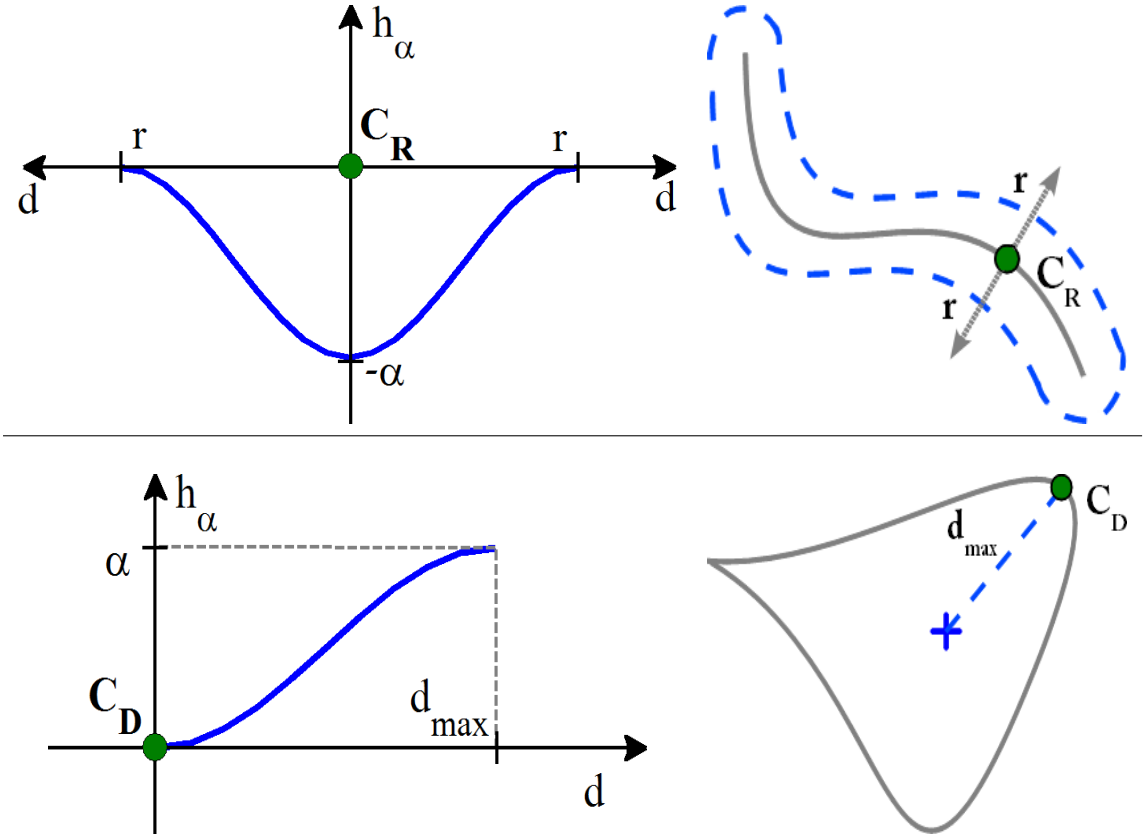


Figure B.8: Top: the function h_α representing a river section. C_R is a arbitrary point along the river centreline. Bottom: the function h_α defining delta deposition. C_D is a arbitrary point along the delta contour.

Figure B.8 top left, shows the graph describing the river bed section, given by $h_\alpha(z) = \alpha h(z)$, where $\alpha \in \mathbb{R}$ and $h(z)$ is defined as $\frac{sin(\pi z - \frac{\pi}{2}) - 1}{2} \forall z \in [0, 1]$, 0 otherwise. Here $z = \frac{d}{r}$, where d is the distance to the centreline and r is half of the width of the river. d is defined to be the minimum of the distances to the segments of line defining the curve. Since $d \geq 0$ and $r > 0$, $z \in [0, 1]$ is equivalent to the more intuitive relation $d \leq r$; i.e. the height h is inversely proportional to the distance d when the considered point of the grid $\mathcal{G}$ (relative layer) is inside the neighbourhood defined by r. Outside this neighbourhood, the height is set to zero.

During the erosion process along the centreline, there is no need to evaluate the distance for every point of the grid that defines the relative layer. We consider a subset S_i of the grid $\mathcal{G}$, in which to check the amount of erosion to apply, that is the union of two squared subset of $\mathcal{G}$, S_i^A and S_i^B, centred respectively at the two limit points A and B of the i-th segment of the centreline, and with half-side length $L = \sqrt{2}T, T = \frac{||A-B||}{2}$.

Modelling a Delta When the first and last point of a curve are close enough (according to a predefined threshold), we consider the curve as closed. A closed curve is interpreted as the outer boundary of a delta as seen from above (map-view). The user can also specify a scaling factor to control the height of the delta. In the following, we introduce a mathematical description of our delta sections (shown in Figure B.8 bottom).

This time, the image of function $h : [0, 1] \longrightarrow [0, 1]$ has positive values, since it implies a deposition, resulting in a formation of a dune in the delta region. Cross-sections of deltas are described by the function $h_\alpha(z) = \alpha h(z)$, where $\alpha \in \mathbb{R}$ and $h(z)$ is defined as $\frac{sin(\pi z - \frac{\pi}{2}) + 1}{2} \forall z \in [0, 1]$, 0 otherwise. Here $z = \frac{d}{d_{max}}$, where d is the distance to the contour and, different from the river case, d_{max} is the maximum distance to the contour of all the points of the grid $\mathcal{G}$ inside the contour. Once more, since $d \geq 0$ and $d_{max} > 0$, $z \in [0, 1]$ is equivalent to $d \leq d_{max}$, but now the height h is directly proportional to the distance d to the contour of the delta when the considered point of the grid is inside the contour. Outside the delta boundary, the height is set to zero.

During the deposition process inside a delta contour, we consider a subset S_i of the grid, in which to check if a point is inside the boundary of the delta or not, that is centred in the i-th point V_i of the contour C and has half-side length

$$L_i = \max_{W \in C} ||V_i - W||.$$

Modelling Mountains, Lakes and Constant Layers There is the possibility to create geological shapes other than rivers and deltas, by building on the previous definitions. Using the delta definition with negative values, we can erode the shape of a lake or basin. The user can also specify if an additional deposition of the same amount as the erosion should be applied to effectively fill up the hole again. The fill will be a unique layer that can be assigned material properties. For instance, one can use transparency to give the effect of a water-filled lake. In such a way, lakes or valleys can be added to the model using a negative scaling factor (as in Figure B.9), or mountains with a larger amount of deposition (as in Figure B.10). Deposition taking place over water-filled rivers or lakes will act naturally on the water by sinking to the bottom. In practice, this is realized by re-arranging relative layers so that deposit occurs before the relative layer consisting of water. Then the water layer is reduced in height by the amount of sunk material, but never more than its depth. This enables us to place the small islands appearing in the river in Figure B.2 right. In addition, layers of constant, user defined thickness may be introduced in the layer-cake.

B.4 Results

The examples used in this paper are generated through a sketch-based technique to acquire a shape definition of the different relative layers, as in Figure B.9, where the shape of each lake is defined by a single user stroke.

Several geological features can be generated with our method. A river with its sedimentary history can be achieved by drawing the initial and the final configuration of its shape evolution, as illustrated in Figure B.3. A delta representation is obtained by sketching a closed curve. Figure B.1 contains an example of a delta. The grain size of delta depositions is dependent on the distance to the mouth of the river which deposits the material. Heavy particles fall down first, while smaller particles are suspended in the flow for longer distances. Our system allows the user to describe this feature as shown in Figure B.7. A colour transfer function is used to map distances to colours. Lakes and empty basins (as shown in Figure B.9) are simple shapes that are not always relevant for exploration purposes, but important to give a context to an illustration. Mountains can be made to improve the context of the illustration (Figure B.10) as they are often

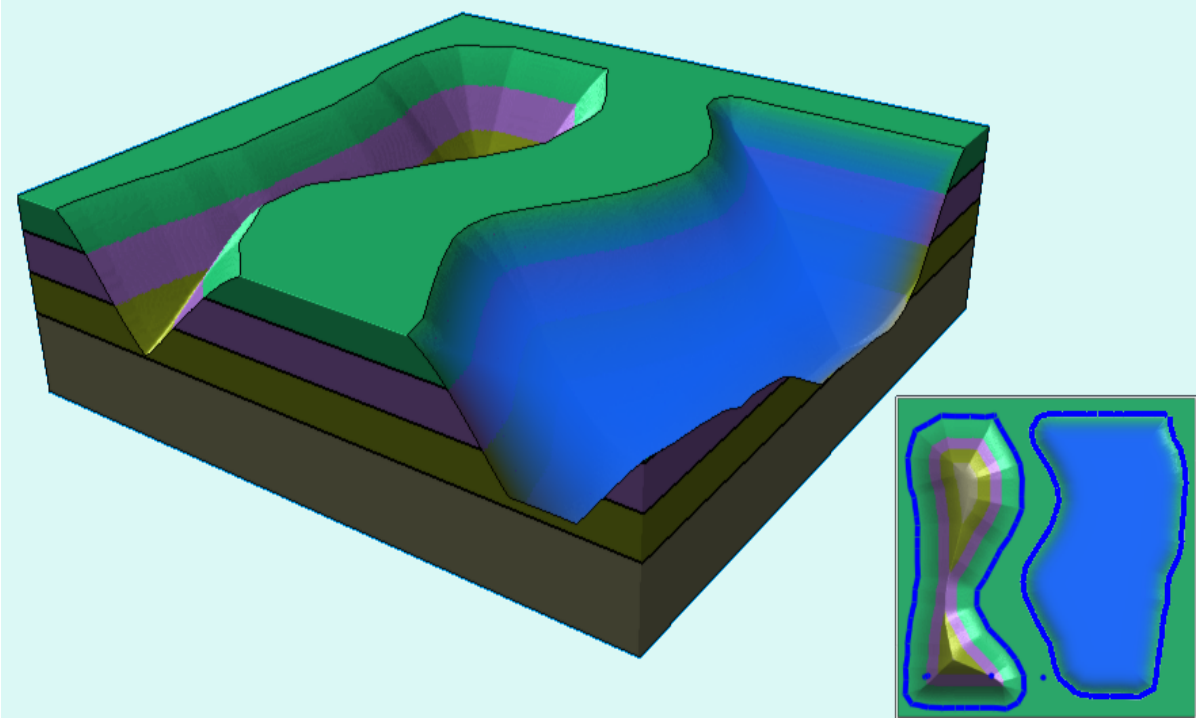


Figure B.9: Two areas eroded into three layers of constant thickness. The rightmost area was filled with material, which has been assigned a translucent water appearance.

the source of depositional material. A composition of all the previous features produces a global overview of a geological scenario, as depicted in Figure B.1. To show that our method can create results comparable with illustrations, we reproduced a detailed 3D version of an illustration describing fluvial evolution as demonstrated in Figure B.2.

An important characteristic of our model representation and visualization is the ability to incorporate intermediate stages of erosion and deposition, that define the sedimentary evolution of a geological scenario. This is important for fluvial systems which are central to understand for geologists. The illustration on the left side of Figure B.2 was made to capture the different depositional characteristics of a river system from a proximal to distal position. It has been made in conventional 2D vector graphics software based on drilled wells and interpreted river footprints found in 3D seismic datasets in the Barents Sea ([66]). Such drawing processes are time-consuming, and in this particular case the illustration took a full day's work to complete. The resulting block diagram is also implicitly static, which entails that upon finalizing, it cannot be easily altered. It took one hour to reproduce the illustration with our approach giving a fast, automated and interactive way of drawing the diagram making subsequent alterations less laborious. Cutting into the model additionally gives the illustrator a unique opportunity to cross-reference her illustration and quickly show the internal implications of external shapes (e.g. the internal shape of a point bar drawn from above can be examined at various angles and cuts, as shown in Figure B.3 and Figure B.11). Drawing schematic block diagrams in conventional illustration software is today a natural approach when investigating 3D seismic data, consequently our technique could be highly relevant for illustrations based on such data. Our geologist co-author pointed out that with our approach, the user benefits from an interactive, on-the-fly method of drawing, instead of having to run through different processing steps (in 3D software) or meticulous

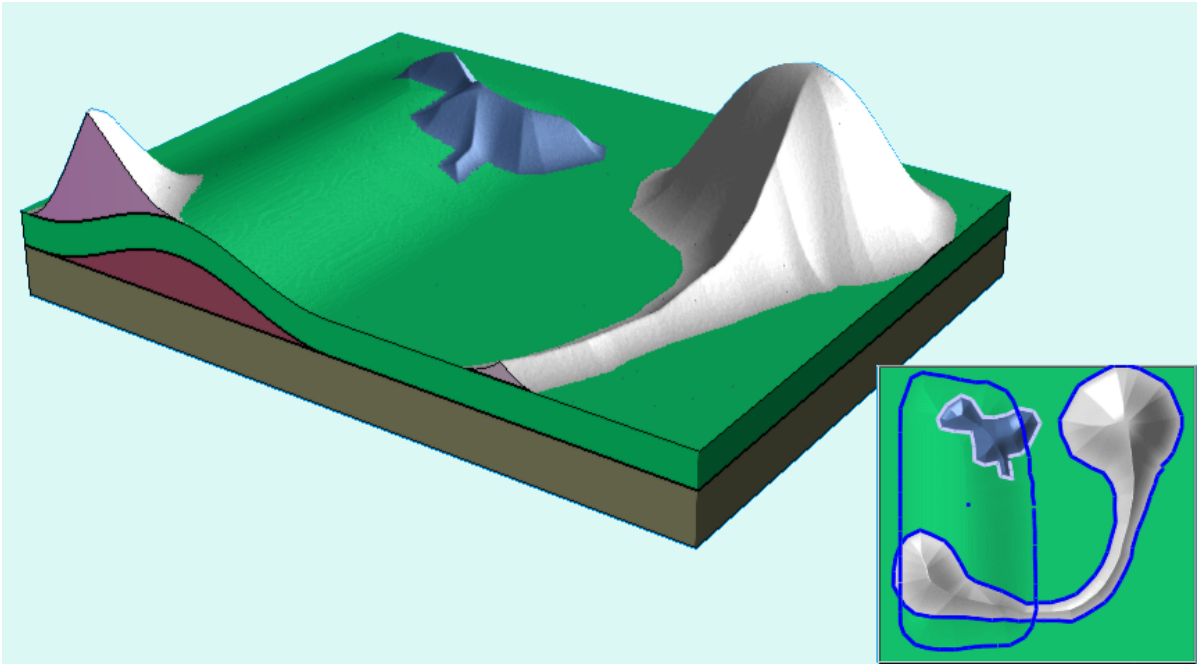


Figure B.10: Layer-cake model with few elements. The distance-based height calculation results in mountain-like features with a varying mountain ridge height.

block-diagram drawings (in 2D drawing software). Figure B.11 demonstrates the ease of drawing time-series illustrations with our approach by adding additional strokes to an initial model. Images on the left side appear in a physical geology book by Thompson and Turk [131]. We created the first time step by drawing two rivers using two strokes and then additionally one stroke each for the following three images. The bottom-right corner is a magnification of a cut-view of the last illustration showing each deposition with individual colouring.

A model consisting of 20 layers is rendered at 40 frames per second in a window of size 1200 x 600. Procedurally generating the 20 relative layers and transforming them to absolute layers takes approximately 3 seconds with our unoptimized code. Timings have been performed on an *Intel Xeon E5620* CPU (12GB of RAM) with an *NVIDIA GeForce GTX 580* GPU (1.5GB of dedicated memory).

B.5 User Evaluation

To evaluate the usability and generality of our tool, we performed two separate user studies on domain experts. One study consisted of having 6 geological domain experts (users A-F) individually watch and give feedback on a tutorial video (accessible here [88]). The video demonstrates how to create each of the geological features supported in our approach followed by a more complete example in the end. Feedback was given by means of grading a list of statements (Table B.2) about our approach and by giving additional comments. Results from the gradings are shown in Figure B.12 a). All participants have or are taking a degree in geology. Users A, B, C, D, E and F are respectively a Master's student, PhD student, PhD student, PhD student, university researcher with PhD degree, geologist at Norwegian Petroleum directorate with Master's degree.

In our other study, an expert in the tool individually gave a different group of 4 users a

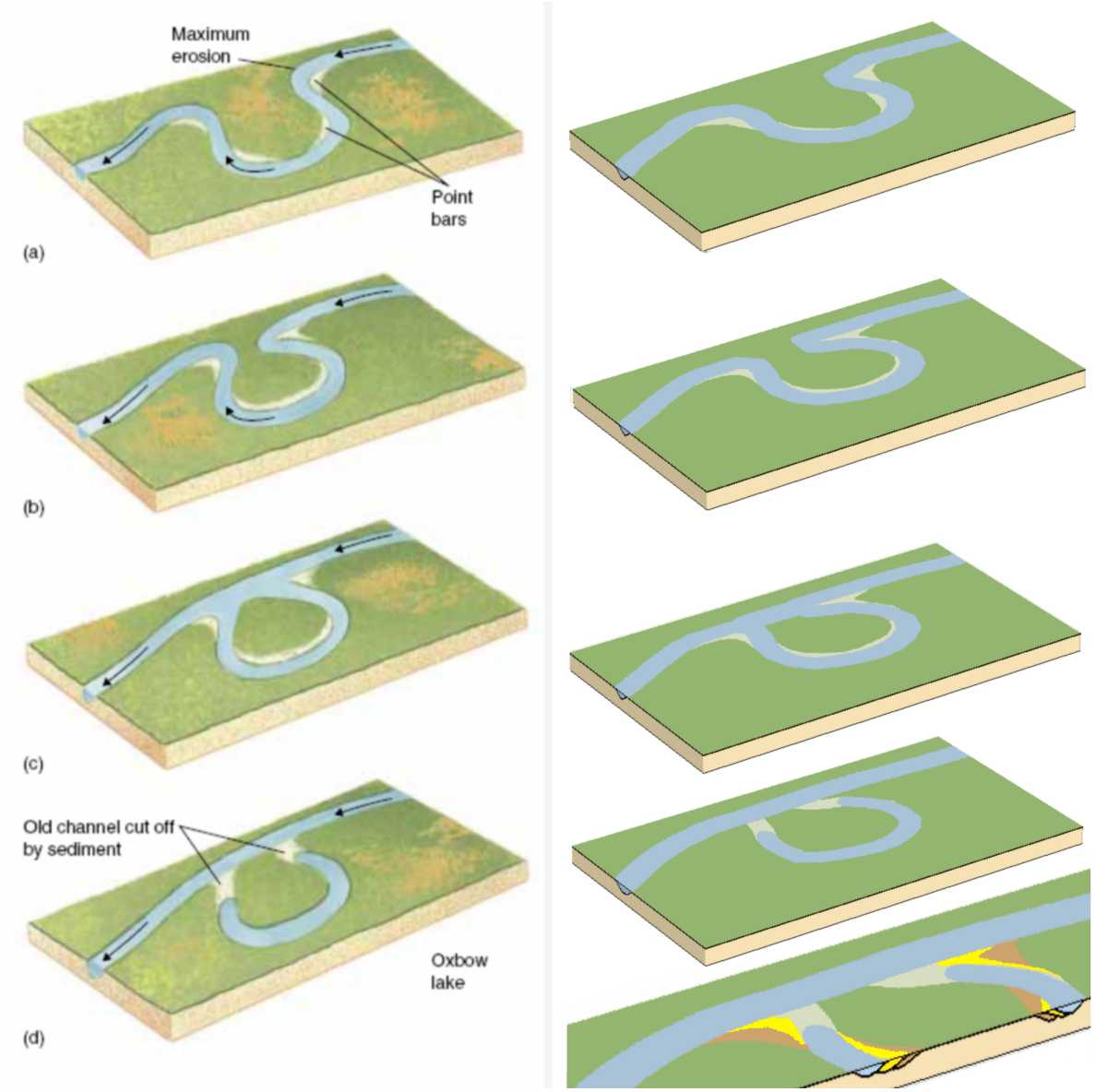


Figure B.11: Left: sequence of illustrations showing the evolution of a meandering river with oxbow shape by Thompson and Turk [131]. Right: Reproduction with our approach requiring only five strokes.

tutorial and then let them use the tool for about 45 minutes. During the session, the tool expert was present to provide assistance, for discussions and answering questions. After each session, the participants rated the same statements as in the first study. Results are shown in Figure B.12 b). In addition, the tool expert noted comments that arose while using the tool. Users G, H, I and J have respectively the background of professional illustrator, post-doc in mining geology, post-doc in sedimentary geology, professor in sedimentary geology.

Some general observations are that all average scores within each question were higher than 0, indicating an overall positive feedback, and all individual scores were 0 or higher, except for a score of -1 given to question 10 by user A and question 12 by user J which we will elaborate on. Also, participants who were able to test the tool scored each question equally or more highly than participants who only watched the video, except for question

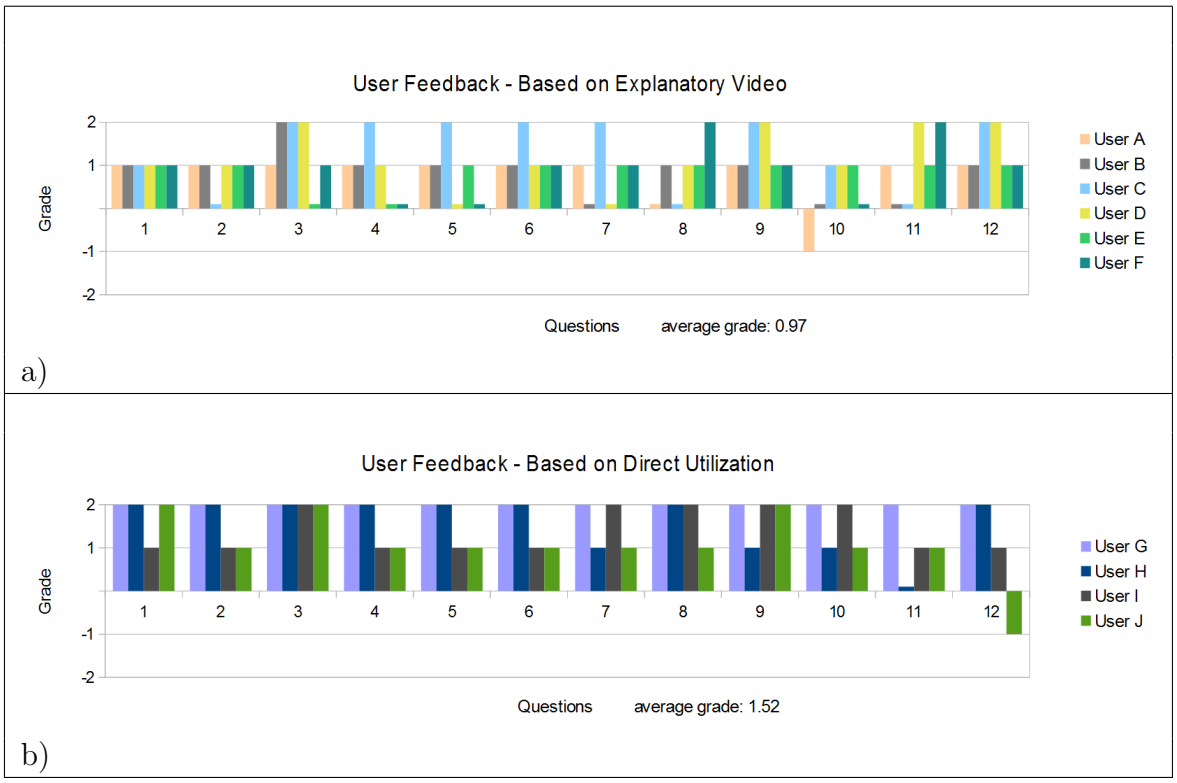


Figure B.12: User feedback on our approach obtained from domain experts.

12 which we will also elaborate on.

Questions 1-7 attempted to quantify the user-friendliness of our approach. Scores were mostly positive with a few neutral indicating that the tool is easy to use. (Survey a had minimum 1 on question 1 and 6 and minimum 0 on the others, however all with average closer to 1 than to 0. Survey b had minimum 1 on all these questions.). Questions 8-12 attempted to quantify the usefulness of the approach in terms of the ability to rapidly create geologic illustrations (question 8); advantages of 3D volumetric illustrations over 2D static illustrations (question 9); the expressiveness of the final illustrations (question 10); the ability to use the results for communicating reasons (question 11) and for teaching purposes (question 12). Question 8 had 8 positive scores and two neutral which can be considered as good, while question 9 had all positive scores. Question 10 had positive scores from all of the utilizers of our method, worse scores were given by the ones who only watched the explanatory video, although the average score for them was positive. User A, a Master's student, gave a negative score based on the argument that if someone is very good in making 2D illustrations, then our 3D modelling approach will be less expressive and might even be slower. Question 11 had 7 positive and 3 neutral scores which could be considered a good evaluation. User H gave score 0 based on that the approach seemed powerful for sedimentary systems, but does not go beyond that in terms of modelling features of structural geology such as faulting. Finally, question 12 had 9 positive and one negative score. User J, a Professor, gave the negative score because of the restrictions with our approach focusing on specific features and requiring a computer compared to a classical classroom setting for a teacher that is trained at drawing geological illustrations on a blackboard. We also received several positive comments with respect to the usefulness of our approach. One commented that "The system seems really useful if you study a fluvial system behaving in a regular way". Several commented

Scale is -2 = very bad, -1 = bad, 0 = neutral, 1 = good, 2 = very good.
1. The approach is fast to learn how to use.
2. The approach is easy to use.
3. It is simple to sketch layers of constant thickness.
4. It is simple to sketch deltas/mountains.
5. It is simple to sketch lakes.
6. It is simple to sketch rivers.
7. It is simple to sketch sequences of intermediate rivers.
8. The approach allows for rapid sketching of geological illustrations.
9. 3D rotation and volumetric cutting into the model improves understanding.
10. The geological features that the approach attempts at expressing are satisfactorily reproduced in the illustrations it creates.
11. The approach provides support for communication.
12. The approach is suitable for teaching purposes.

Table B.2: Statements graded in the user study.

that there is a lot of potential in our approach both for visualizing systems in 3D and for speeding up the process of creating illustrations. Finally, volumetric illustration allowing cuts through the volume was appreciated. The user study indicates that our approach is user friendly and that it covers most of the structures relevant for fluvial systems and depositions but is too limited for more general geology such as structural geology which however was not our focus.

Suggestions for further improvements were the possibility to export models to vector-based illustration software to be able to add finishing touches. Two users expressed the wish of having a coordinate system that would allow them to use measurements as input.

B.6 Conclusion

We have identified sketching needs by geologists who model fluvial systems based on interpretations. To address these needs we have constructed a sketch-based interface, a layered data representation and a rendering approach. The data structure leads to an intuitive definition of the geological process of deposition and erosion and requires only basic arithmetic calculations for realizing the model. This results in a volumetric model which relieves us from topology and intersection testing required for boundary representations. This work has been performed in tight collaboration with a geologist defining the problem domain, and who is a co-author of the paper.

Limitations Due to our compact representation of the model using $2D$ grids, a layer cannot fold back onto itself. Hence, we are not able to create geological scenarios with overhangs. With respect to depositions in general and specifically rivers and deltas, this is not a major problem as they originate from erosional and depositional processes that mostly work in the vertical direction.

Our approach focuses on fast creation of illustrations for communication purposes and unconstrained hypothesis testing. Therefore we did not build in constraints and elements of physics simulation as this can burden the user with complex input parameters and long processing times. It is up to the user to honour geological and physical constraints such as unrealistic terrain gradients and slope stability.

Future Work As tectonics, subsidence, slope failures and faults can be handled by our data structure with small modifications, we plan to include these geological processes in a next prototype of our sketch-based system. The calculation of the relative and the absolute layers is parallelizable and could in a future version be performed on the GPU.

Another improvement to our models would be given by including the technique of Zhu et al. [148] to enrich channels with animated visualization of flow.

Finally, 3D models obtained with our approach that include some level of heterogeneity, as suggested with grain size trends in deltas (Figure B.7), could serve as training images for *Multiple-Point Statistics* (MPS) (Caers [21], Strebelle [128], Arpat and Caers [7]).

Acknowledgment

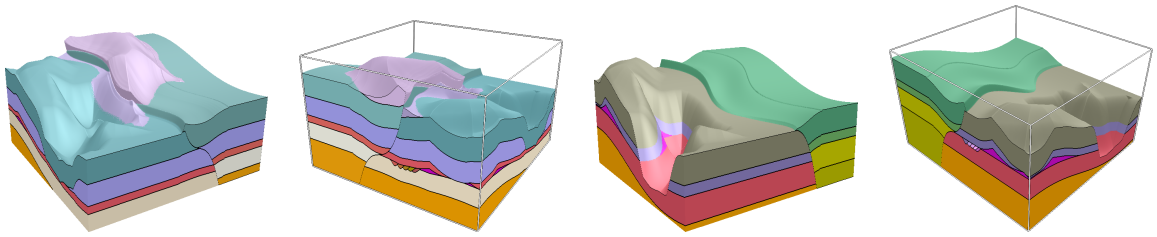
The authors would like to thank all reviewers that gave comments and suggestions for improvements, and Ivan Viola for fruitful discussions. This work is funded by the Petromaks program of the Norwegian Research Council through the Geoillustrator project (#200512).

Paper C

Rapid Modelling of Interactive Geological Illustrations with Faults and Compaction

Mattia Natali[1], Julius Parulek[1], Daniel Patel[2]

[1]University of Bergen, Norway
[2]Christian Michelsen Research, Bergen, Norway



(a1) (a2) (b1) (b2)

Figure C.1: Illustrative geological models showing non-planar fault and compaction of layers: (a1)-(a2) and (b1)-(b2) are different views of the same model.

Abstract

In this paper, we propose new methods for building geological illustrations and animations. We focus on allowing geologists to create their subsurface models by means of sketches, to quickly communicate concepts and ideas rather than detailed information. The result of our sketch-based modelling approach is a layer-cake volume representing geological phenomena, where each layer is rock material which has accumulated due to a user-defined depositional event. Internal geological structures can be inspected by different visualization techniques that we employ. Faulting and compaction of rock layers are important processes in geology. They can be modelled and visualized with our technique. Our representation supports non-planar faults that

This article was published in *Proceedings of the* 30^{th} *Spring Conference on Computer Graphics (SCCG)*, and presented at SCCG 2014 in Bratislava, Slovakia by Mattia Natali.

a user may define by means of sketches. Real-time illustrative animations are achieved by our GPU accelerated approach.

C.1 Introduction

For many years the focus of modelling has been on achieving an accurate representation of the real world. This has led to complex techniques that produce precise models, but demand considerable working time of specialized personnel. Rapid modelling and illustrative visualization [137] help reduce such efforts. Rapid modelling can be carried out either in form of automated procedures or through sketches that are interpreted by specialized algorithms. Sketch-based modelling is a way of making a new model or modifying an existing one, with the help of human drawing ability. Sketching is more intuitive for a designer than setting parameters. Sketch interpretation is closely tied to a particular modelling scenario, but it is basically an automatic way of filling in parameters. A sketch-based approach has advantages compared to procedural modelling in terms of expressibility: with the former, a user can freely design shapes and this is usually more intuitive and simpler to learn. Some techniques interpret sketches as if they were made on a sheet of paper [81, 63], hence they attempt to bring back to 3D space what were meant to be projections of the model. Other techniques allow to directly define strokes in 3D [14, 60, 48, 139]. It is useful to combine sketch-based modelling with illustrative visualization, because sketches are meant to be approximations and this is best conveyed through a style that relates to qualitative rendering.

In illustrations, details are less important, while the focus is on the concept a model should convey, rather than the preciseness of representing data measurements. Illustrations are mainly utilized for communication and teaching purposes. In our work, we merge sketch-based modelling and illustrative visualization to produce geological models (such as the example in Figure C.1). They are represented as layer-cake objects that show the internal stratigraphy of the subsurface. We address the modelling and interactive animation of geological aspects, such as faults and compaction of layers.

Currently, illustrators in geology spend a considerable amount of time in designing their models in 2D vector graphics software. That results in two-dimensional images that convey three-dimensional structure. Such models cannot be rotated in 3D, cut into or animated without enormous efforts. Animations are particularly important in the domain of geology to communicate variations and combinations of processes that take place in the subsurface. Our approach allows to quickly define illustrative animations of some of the most relevant geological events involved in the subsurface (as shown in Figure C.2, where we use our method to quickly reproduce key-frames of a fault animation available on the web [1]).

Central geological events that shape the subsurface are deposition, erosion, folding, faulting, igneous or salt intrusion and layer compaction. While there has been research progress in rapid modelling of stratigraphy formation (deposition and erosion) and folding [89], there are no rapid modelling methods for faults and compaction events.

Faults are important geological features. Their interpretation leads to an understanding of the behaviour of the crust of the Earth. Movements in the crust produce faults in rock layers. For instance, a standard approach to derive the direction of two lithospheric plates is to study faults generated by their displacement.

In summary, our contribution is an improvement to the work by Natali et al. [89] with the addition of interactively animated faulting processes and compaction effect, due to deposition of upper layers. Moreover, we employ different visualization techniques that facilitate a visual navigation through the stratigraphic model, such as exploded view and adaptable staircase view. Computations that handle the processes and model

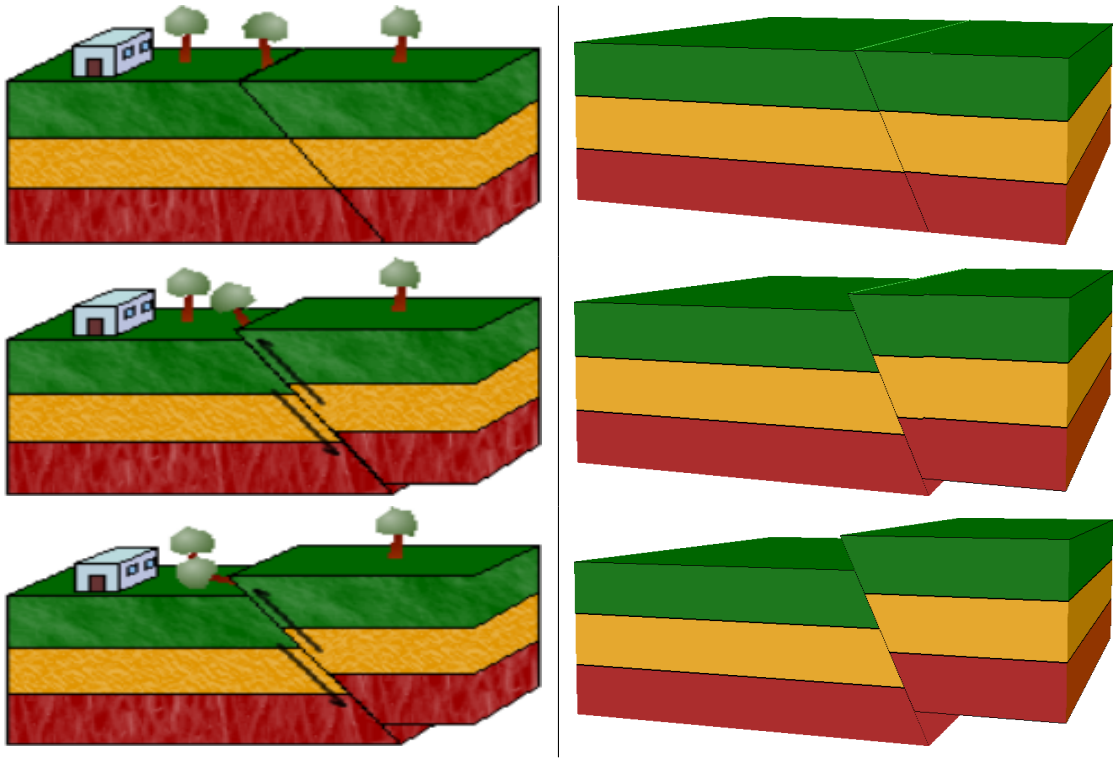


Figure C.2: Key-frames of a hand-made fault animation next to key-frames of our animated technique.

representation are carried out on the GPU. In this way, we are able to interact with faulted models composed of many layers and animate them in real time.

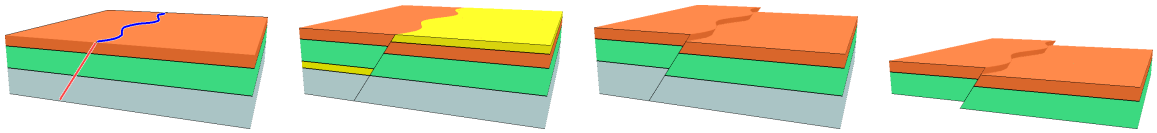
C.2 Related Work

For a wide overview on illustrative visualization techniques and their application to several fields, we suggest the tutorial by Viola and colleagues [137]. When considering medical illustrations, Sousa et al. [124] present a volume illustration method for interactive simulation sessions. While in geological illustrative visualization, Patel et al. [104] propose an approach to display volumetric seismic data.

Sketch-based modelling has evolved in recent years, together with interpretation of 2D curves for 3D reconstruction. For instance, the work by Zhu et al. [149] achieves illustrations of scientific concepts, that are then enriched with parts animated by 2D flow simulation. Similarly, the paper by Rivers et al. [118] enables users to generate complex 3D models with sketch-like input. Owada et al. [98] present an interactive tool for modelling and inspecting volumetric illustrations with textures.

Sketch-based modelling of terrains has been addressed in the last years by a handful of works [48, 57, 90]. Nevertheless, all of them aim at obtaining a terrain surface, they do not focus on the subsurface or geological features in it. What is really important for geological illustrations is the internal structure. That is: stratification (deposition and erosion), folds, faults, compaction of layers, fluvial systems and salt domes.

If we look for illustrative visualization in stratigraphic geology, the work by Natali et al. [91] and an extension to produce animation, proposed by Lidal et al. [73], can be found in literature. Both methods are able to include faults, but one limitation is given



(a) (b) (c) (d)

Figure C.3: Steps needed to obtain a reverse fault are shown in this figure: from the sketches in red and blue defining the fault surface (a) to the final illustration (d). (b) and (c) are intermediate steps to displace the fault blocks, respectively with and without highlighting the supporting wedges. In (c) the top wedge is transparent and the bottom wedge is given the colour of the bottom blue layer.

by the fact that each model is an extrusion of 2D curves to 3D space. Therefore it is not possible to define internal structures or faults which have other than linear edges when seen from aerial view. In the former work [91], two-dimensional sketches are drawn on a cross-sectional view and interpreted as boundaries of the layers or as fault surfaces. The user has to define the shape of the folded layers, a 2D fault and its displacement. Sketches correspond to surfaces that are then triangulated to build the layer-cake model. In the latter work [73], time is taken into consideration as well. Key-frames are drawn on cross-sectional views. Interpolation between key-frames leads to an animation of the sketches, and thus of an extruded 3D model. On the other hand, with our approach, we are able to define internal structure, because the modelling process happens sequentially. Each stratigraphic layer represents a deposition at a certain period of time. Similarly to nature, where depositional and erosional processes generate historical imprint of the rocks, every following layer (partly) covers the one below. The idea of using layered data was initially proposed by Benes and Forsbach [11], and subsequently extended in other works [92, 89].

Looking at the state of the art of geological modelling [90], few approaches can in theory be adapted to produce animation of faulting. One possible solution is given by representing the layer-cake model by means of meshes (either two- or three-dimensional). The disadvantage in this way of proceeding is given by the continuous computations that are required to obtain all the intersections between each surface and the fault. Every small displacement in the faulting process involves a re-computation of intersection points. To compute the intersection between two meshes is already an expensive task, if a fault passes across many layers, the computational cost increases. This approach would require longer processing time compared to our rapid modelling. For simple planar faults, a mesh of the model can be divided in two by the fault plane. When this subdivision has been calculated, each part can then be slide along some user defined vector on the fault plane. No recalculation of intersections needs to be performed, and none of the parts need to be deformed due to the planarity of the fault. If the model is represented by Constructive Solid Geometry (CSG), faults can be made by clipping the model into smaller pieces. However, each piece would not deform, because only affine transformations are supported.

In literature, a further technique would be available to generate fault illustrations, also for non-planar faults. This technique is known as illustrative deformation, as introduced by Correa et al. [31]. Models are defined by 3D volumes. The shape of a fault and its animation could be described by a predefined template. This would require that an expert has to author a set of templates in an authoring stage, one for each type of fault to be illustrated. Authoring a single template is time demanding, for communicational

or teaching purposes many templates would be needed. Furthermore, every template has high memory requirements, much more than what is required by our compact layered representation.

C.3 Outline

In this section we give a complete overview of the process that allows us to generate a geological illustration from scratch, as shown in Figure C.3. We start from an empty box in three-dimensional space, that, as a container, bounds the volume of the model we will define. A geological event is described by a single sketch (either a point or an open or closed polygonal curve) and sketches are drawn on top of the box, as shown in the left image of Figure C.4 where the box is viewed from above. For defining faults, the user also draws on the side surfaces of the box. Our system directly deduces what shape to create from the sketch: a single point means a layer of constant thickness (for example, Figure C.3 (a) is initially defined by three layers of constant thickness); an open curve defines a river (as the upper sketch in Figure C.4); a closed curve defines an area of erosion or deposition (as the lower sketch in Figure C.4); a curve on a side of the box, followed by an open curve on top, defines a fault (an example of this case is shown in Figure C.3 (a) by the red and the blue curve respectively). After having sketched a structure, the user can set parameters such as thickness of constant layers, degree of deposition/erosion or depth of rivers. Each map-view sketch is interpreted and converted to a heightmap, as shown in Figure C.4. The superimposition of the heightmaps produces the layer-cake model, as described by Natali et al. [89]. The order in the sequence of sketches is important, because each corresponding heightmap is interpreted as a specific geological event in time. The sketches, each representing a depositional or erosional event, are sequentially stacked. The conversion from sketch to heightmap requires a computation of a distance field on a grid, i.e. the distance of a grid point from the polygonal curve describing the sketch. This step is done only once during the model construction.

When a fault intersects a layer, the layer is split in two parts and saved as two distinct heightmaps, as described in Section C.4.2. The displacement that is caused by the fault is defined by the user and assigned to the layers intersected by the fault (only one of the two layers obtained by the splitting moves according to the fault displacement). If more than one fault acts on the same layer, the displacements associated to each fault are summed together. The layers are only displaced in the rendering stage. Animation of a fault is obtained with a slider that reduces/increases the displacement associated to the fault.

We also simulate compaction. As the number of layers increases, the covered layers decrease in thickness as a function of the amount of material above.

C.4 Modelling Approach

We briefly describe how the model representation is defined in Section C.4.1. We then describe the procedure that is necessary to perform faulting and compaction in Section C.4.2. In Section C.4.3, we propose three types of illustrative visualization techniques that can be employed to better show internal structures. The use of one visualization technique is not exclusive for the others, they can be combined. Finally, Section C.4.4 identifies parts of the algorithm that have been parallelized and implemented

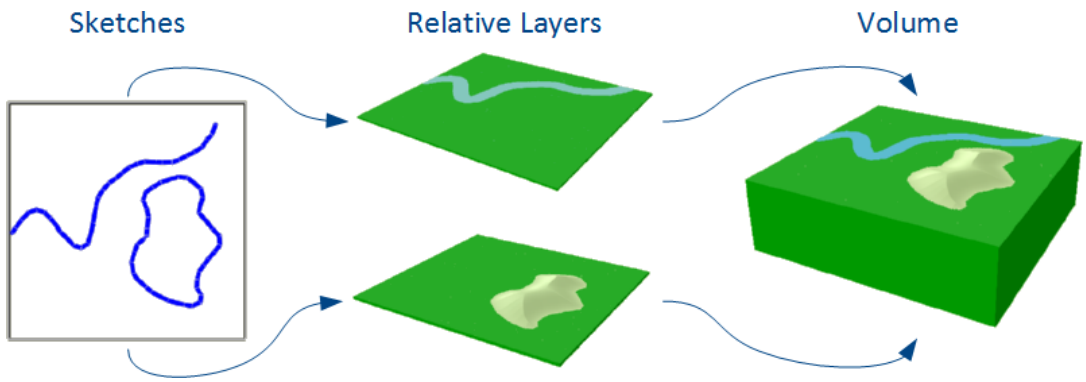


Figure C.4: Overview of the conversion from sketches to relative layers first, and to volume representation afterwards.

on the GPU to achieve interactivity.

C.4.1 Internal Representation

We use the simple and effective layered representation described by Natali et al. [89] for our 3D models. We have improved the method with respect to computational time involved in both the processing and the rendering stage, by parallelizing it on the GPU. This is necessary to create real-time animations. The approach presented by Natali et al. encodes the layer structure as a stack of heightmaps, each representing an event of deposition or erosion. Each heightmap therefore represents a geological event that happened at a specific time. In this paper, we generalize this representation to cover a larger number of cases that can be found in a geological scenario.

Each model is based on a dual representation: it uses two alternative types of heightmap, the so called *absolute* layers and the *relative* layers [89]. The two representations have different advantages. Relative layers are used in the definition of the model and computations on it. They are good for keeping track of layers independently from each other in terms of thickness; as well as for faulting by splitting relative layers, and for sliding the faults by offsetting the relative values. On the other hand, absolute layers are employed to generate and render the volume. In addition, the transformation between the two representations can be done in parallel.

The i-th *absolute layer* $h_i^{abs}(x, y)$ is defined for each point $(x, y) \in \mathcal{G}$ by the height of the top surface of the layer in the reference coordinate system of the model, $\forall i = 1, \ldots, n$. Where n is the number of layers, $\mathcal{G}$ is a uniform 2D grid and $h_0^{abs}(x, y)$ is defined to be zero in every point.

The i-th *relative layer* $h_i^{rel}(x, y)$ is defined for each point $(x, y) \in \mathcal{G}$ by the displacement

$$h_i^{rel}(x, y) := \Delta h_i^{abs}(x, y) = h_i^{abs}(x, y) - h_{i-1}^{abs}(x, y)$$

of two consecutive absolute layers, $\forall i = 1, \ldots, n$.

The relation between relative and absolute layers let us express the i-th absolute layer as

$$h_i^{abs}(x, y) = \sum_{k=1}^{i} h_k^{rel}(x, y), \forall i = 1, \ldots, n.$$

The above formalization of the relation between relative and absolute layers can be easily proven by substituting the definition of the i-th relative layer into the formula.

C.4.2 Geological Features

Important features that are present in subsurface geology, can be modelled with the representation we employ in our work. Every geological feature in which we are interested, can be defined in our sketching interface using curves. To support fast sketching, our system is, for most situations, able to deduce the type of operation that is sketched, based on how the sketching is performed. In this section, we describe methods to model different geological events.

Deposition and Erosion

As in nature, deposition and erosion are the basic processes that create rock layering. We express them by the sign of the heightmap values. A positive value means deposition of material; a negative value means erosion (of the material deposited by the previous layers). When deposition occurs, a new layer is introduced and its thickness depends on the heightmap values. In case of erosion, layers below are reduced in height, starting from the uppermost and proceeding to the others when their thickness is smaller than the amount of erosion. For instance, the valley visible in Figure C.1 (b1) was created by an erosion which affects three layers, reaching the red one after having consumed and pierced the gray and the violet layer (because in that area the erosion is greater than their thickness).

Faults

A fault is created by splitting each layer that is crossed by the fault surface into two blocks. Since we use a heightmap representation, this corresponds to splitting a relative heightmap into two relative heightmaps, one heightmap for each of the two blocks separated by the fault. For this aim, absolute values of the layers are required, because the fault is defined with absolute values and intersections with layers are absolute as well. Therefore, before applying a fault, we convert the model from relative to absolute layers and we proceed with the inverse process afterwards.

In practice, the whole process of faulting a model is given by the following procedure. All the layers that are crossed by the fault surface must be detected. A fault does not need to entirely cut the model from top to bottom. It is important to be able to also handle the case of so called *blind faults*, when the fault intersects only some of the layers and not the uppermost. For every layer which is intersected by the fault, we compute the points of the heightmap that are affected by the intersection. If we also know the points of intersection of the layer below, it is possible to detect what kind of inclination the fault has. When a layer below does not exist, we have reached the bottom of the model and the intersection is computed with the plane defined by $\{z = 0\}$.

Knowing the inclination of the fault for a certain layer, permits to label each side block as a *foot wall* or a *hanging wall* (as in Figure C.5). This depends on whether the block is respectively self-standing or not, in a hypothetical removal of the counterpart. This further information on the position of foot and hanging wall lets us derive from the direction of movement of the two blocks, if a *normal fault* (divergent direction, as in Figure C.5 (b) and (d)) is created or a *reverse fault* (convergent direction, as in Figure C.5 (b) and (c)). For instance, the example in Figure C.6 is a reverse fault, because the two blocks move towards each other.

We now give an example on how to split a layer into two with a fault. We consider

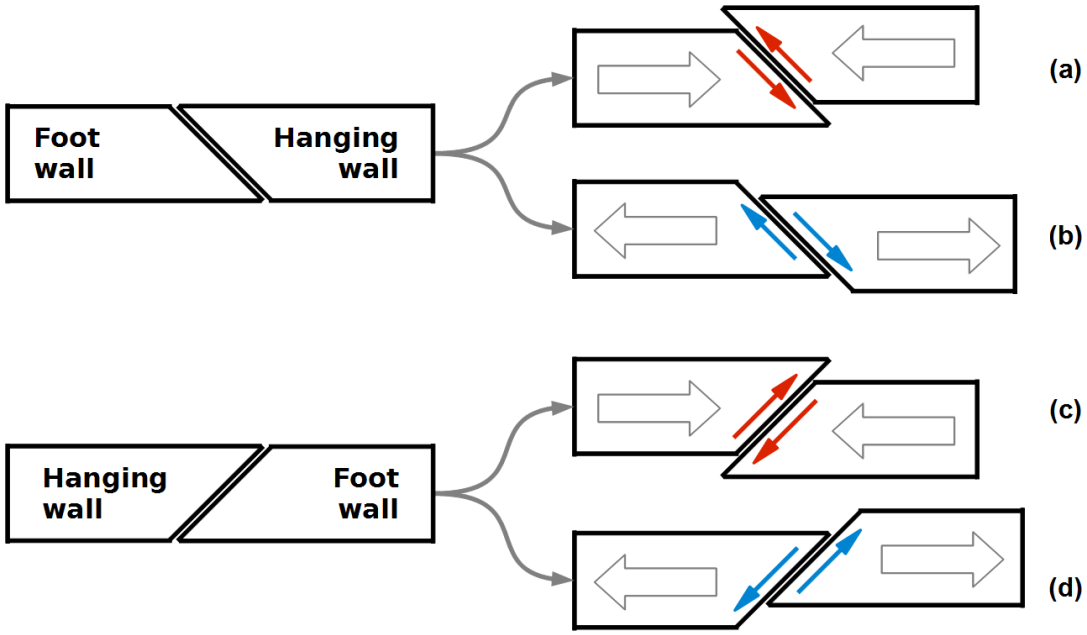


Figure C.5: This scheme exhibits all fault cases our method is able to recognize. (a) and (c) represent reverse faults, whereas (b) and (d) represent normal faults.

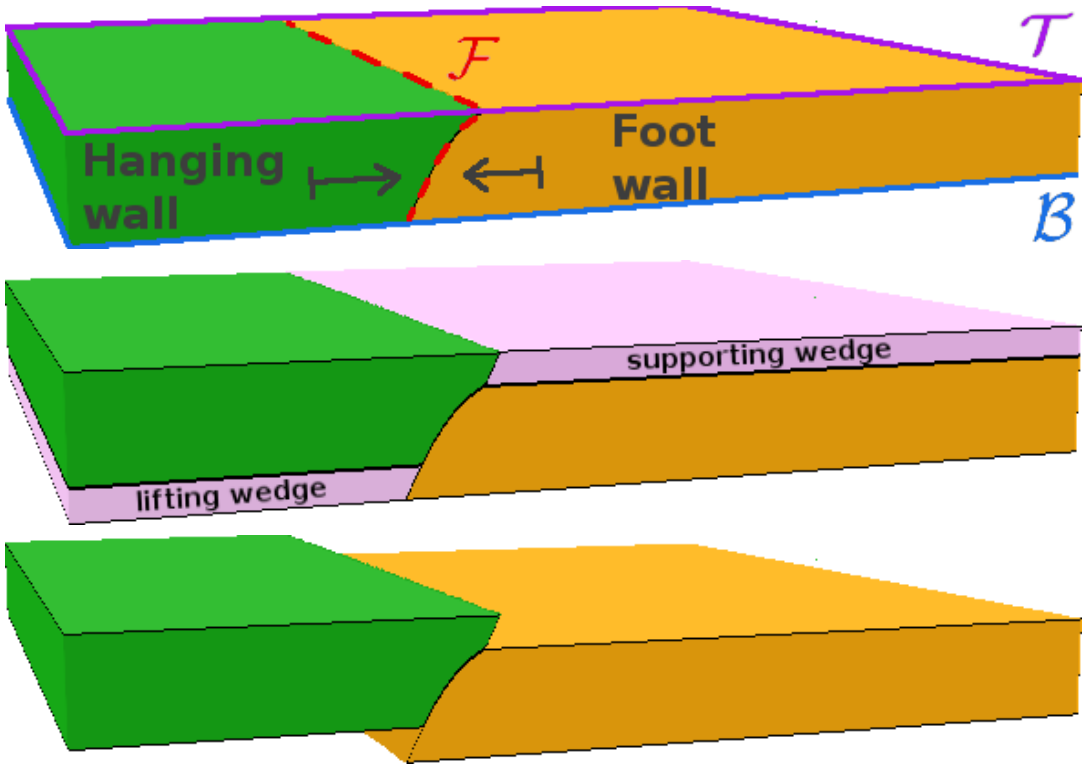


Figure C.6: Procedure that: (top image) detects intersection with the fault surface and separate one layer in hanging (green) and foot (orange) wall; (middle image) inserts wedges (pink) and displaces blocks; (bottom image) sets wedges to be completely transparent.

a reverse fault with foot wall to the right (corresponding to case (c) in Figure C.5), all the other cases (Figure C.5 (a), (b) and (d)) are analogously defined. Notice that the

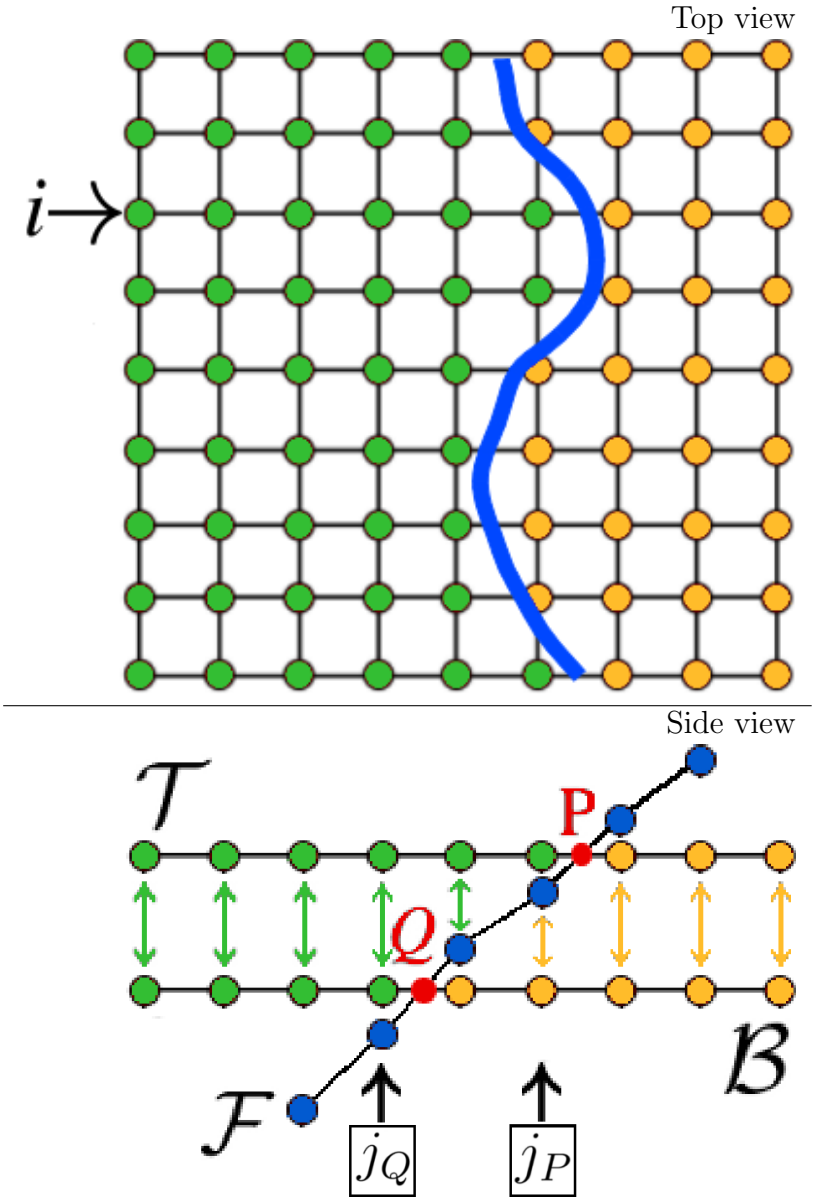


Figure C.7: Top image: grid of the layer that is intersected by the fault (blue curve). Bottom image: cross-sectional view corresponding to the i-th row, where the fault $\mathcal{F}$ intersects the grids $\mathcal{T}$ and $\mathcal{B}$.

process takes place on absolute layers, but output layers (hanging and foot blocks) are relative. This is just a design choice, in order to use the relative layer representation in the subsequent sliding of the fault. The order of relative layers in the stack is important, for instance a foot wall layer must be always before its corresponding hanging wall layer. Let us consider the situation shown in Figure C.6 top, where we have a single layer that is cut by a fault $\mathcal{F}$. The foot wall is assumed to be on the right side and the blocks to have convergent directions. The layer is delimited by a top surface $\mathcal{T}$ and a bottom surface $\mathcal{B}$. We now consider the i-th row of the grid $\mathcal{T}$. An intersection point P between the fault $\mathcal{F}$ and the grid $\mathcal{T}$ is identified by two adjacent grid coordinates j_P and $j_P + 1$ that define the separation of the left and the right layer block respectively (see Figure C.7). Analogously, j_Q and $j_Q + 1$ identify the intersection Q between $\mathcal{F}$ and $\mathcal{B}$. The displacement of the fault is defined by the user and it is a 2D vector V lying on

the plane of the grids, i.e. defined by x and y coordinates. The vector V moves $P_i^{j_P}$ (by our convention hanging wall moves, foot wall is steady) on the surface of the fault. This is done by projecting V on the surface, which returns a 3D vector. This vector is saved together with the points of the grid and it is updated every time a fault is encountered, but it does not affect the grid until all the faults have been applied and the model is ready to be rendered. This avoids computations that would slow down our algorithm and eventually our animations.

The heightmap corresponding to the hanging wall is easily obtained as absolute values, because it has the same values as the top surface that bounds the layer, that is $h^{abs}_{hang} = h^{abs}_{\mathcal{T}}$. The heightmap corresponding to the foot wall, instead, implicitly store the information of the geometry of the fault cut. If we refer to Figure C.7, the foot wall is defined by the absolute value at a grid point (i, j) as follows:

$$h^{abs}_{foot}(i,j) = \begin{cases} h^{abs}_{\mathcal{B}}(i,j) & \text{if } j \leq j_Q \\ h^{abs}_{\mathcal{F}}(i,j) & \text{if } j_Q < j \leq j_P \\ h^{abs}_{\mathcal{T}}(i,j) & \text{if } j > j_P \end{cases}$$

At this stage, we have our data structure ready to display the fault. In case there are other layers on top of the faulted one, they are automatically shifted and deformed due to the sequentiality of the data structure. A hanging wall block moves according to the shape of the fault surface. In case the fault surface is not defined by the user's sketch beyond a certain displacement, we perform a linear extension of the fault surface.

Two hidden layers have to be inserted in the model and we will call them *wedge* layers (depicted in pink in the middle image of Figure C.6). One is needed to lift, if in the presence of a reverse fault, or to lower, in case of a normal fault, the layers of the hanging wall and convey the displacement. In the middle image of Figure C.6, the green block is lifted by the left wedge and its tip is supported by the right wedge. The second wedge is only needed when a reverse fault occurs. It is always transparent and supports the material that otherwise would fall down on the foot wall after the fault displacement. This behaviour is similar to what is found in abstract fault illustrations, like the one displayed in Figure C.2 left.

Compaction of Layers

Rock layers are exposed to a pressure (called *overburden* or *lithostatic* pressure), due to the weight of above layers. Lithostatic pressure increases with depth. In general, porosity decreases with depth, because lithostatic pressure tends to compact empty or fluid filled spaces of a material. To compute the compaction to which each layer is subject (Figure C.8 shows an example before (left) and after (right) compaction), we need to know the total amount of deposition above the layer and the type of rock it is stratified on top. That is, we have to go through the layers on top and detect the height at a specific point (i, j) and its corresponding density of the material.

In a simplified situation, all the layers have the same mass density and it is enough to multiply relative values of the considered layer by a coefficient that depends on the height h^{abs}_{top} (the last layer on top) - h^{abs}_{l} (the considered layer), even for negative values as in Figure C.9.

In this way, the more we deposit on top of a layer, the more it compress and lower its height in the final visualization. In a more realistic situation, a coefficient of compressibility of the material is taken into consideration as well.

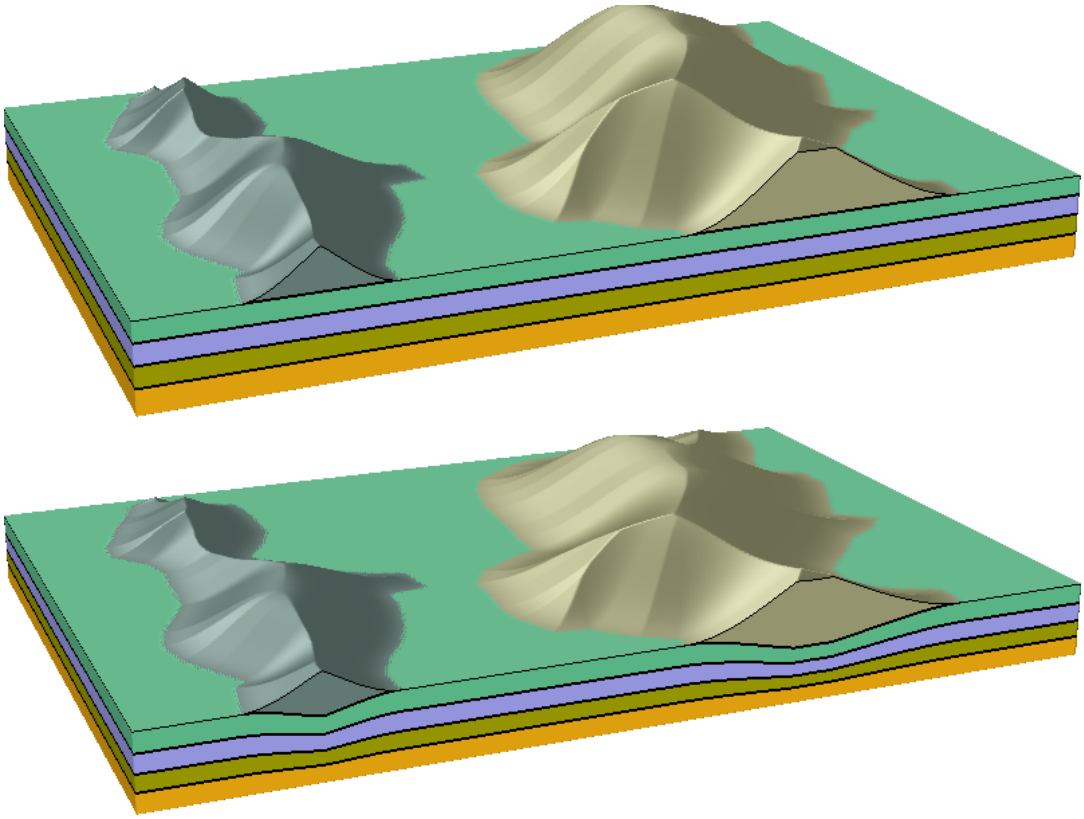


Figure C.8: Comparison between a model where compaction is not applied (left) and where compaction is applied (right).

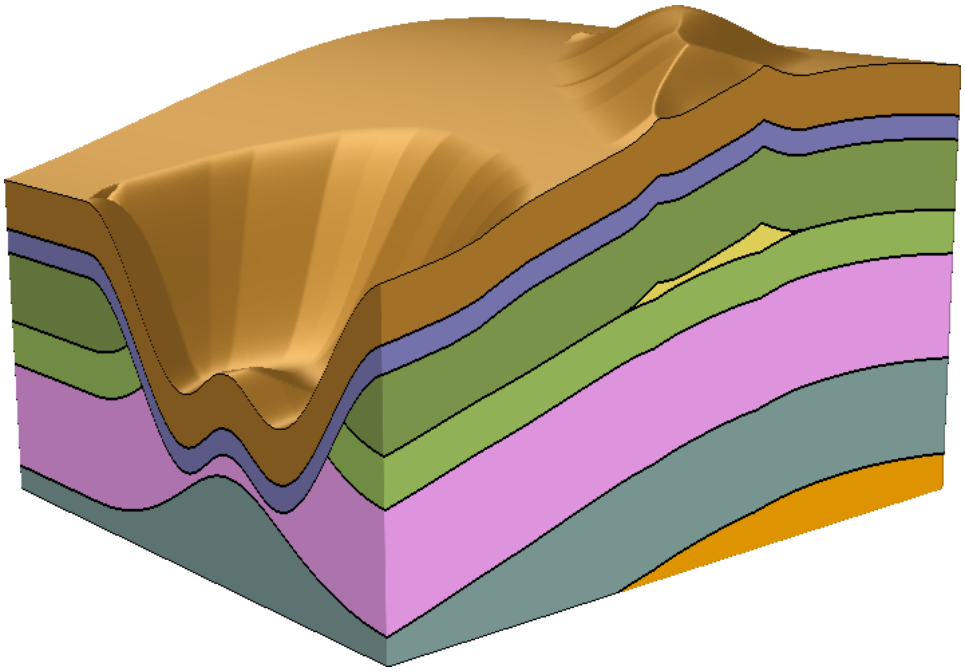


Figure C.9: An exaggerated example that shows in the canyon that our approach produces a decompaction of layers when pressure has been removed from above them due to erosion of material.

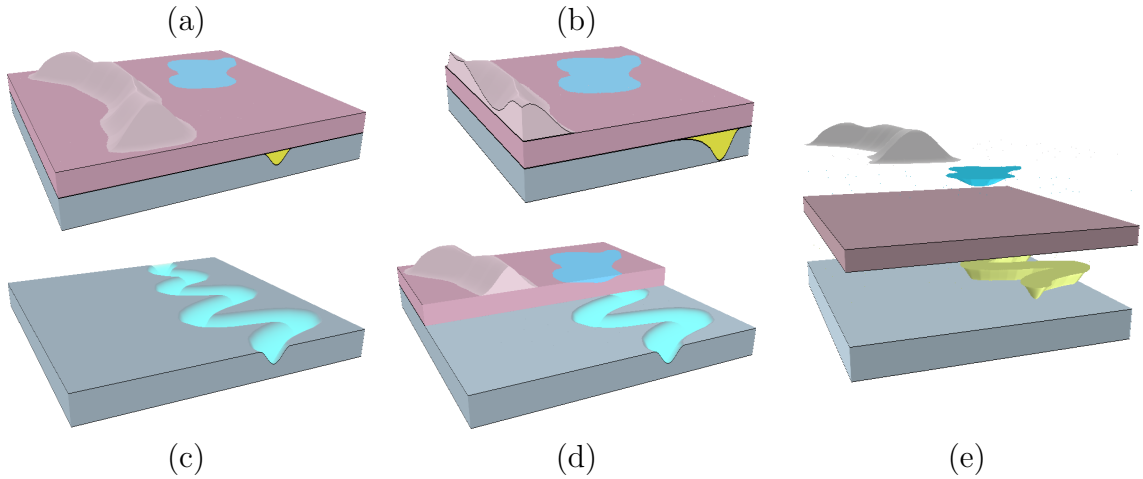


Figure C.10: Different ways of visualizing the same model: (a) entire volume; (b) using two orthogonal and vertical cutting planes through the volume; (c) employing a temporal cut; (d) staircase view; (e) exploded view.

C.4.3 Alternative Visualizations

Cutting planes, as in Figure C.10 (b), are not always ideal for revealing important features in an illustration. We give the possibility to interact with the layer-cake utilizing alternative visualization techniques.

Temporal Slider

In contrast to a cutting plane, which creates a spatial cut, we want to consider a more domain specific temporal cut. Each absolute layer corresponds to a configuration of the terrain in a certain point in time. Therefore, if the aim is to show how the subsurface changed in time, it is enough to slide through the absolute layers by setting only some of them visible. Assuming to have reached a time step t, the illustration is given by rendering only the first t layers $h_1^{abs}, \ldots, h_t^{abs}$.

Staircase View

In this visualization, cutting planes interact with only one layer at a time. A default view employs as many cutting planes as the number of layers, all of them parallel to each other. Starting at the bottom layer and going through all of them, the associated cutting plane removes an increasing part of stratification that is visible (see Figure C.10 (d)).

Exploded View

Exploded views make the internal structure visible by separating each layer. This is achieved by inserting a transparent displacement between layers. Its thickness can be set, which changes the space between layers accordingly. This type of visualization helps to show internal structures, such as channels (see Figure C.10 (e)).

C.4.4 GPU Realization

In the realization of our framework, we build on the representation introduced by Natali et al. [89]. The representation is simple; although powerful enough to tackle all the modelling scenarios described in our paper. Additionally, we improve their method to achieve a real-time modelling performance in order to maintain interactivity when updating the 3D model. We achieve this by migrating parts of the modelling stage to the GPU. Here, we utilize CUDA platform, which realizes a final heightmap construction from a set of relative layers. Each relative layer encodes deposition (positive value) or erosion (negative value), which represents a geological event that happened in a specific point in time. The final heightmap is then allocated, computed and resides in the GPU memory. This allows for direct display by an OpenGL based volume renderer, since the final heightmap is represented as a 3D texture. Here, we exploit CUDA-OpenGL compatibility, which provides us generation of even complex animations at reasonable frame rates (see Section C.5).

The strength of the representation we employ is given by the fact that we do not handle an entire volume during the processing phase, we only consider a limited number of heightmaps, which depends on the number of sketches in the scene. With our boundary representation, we can still simulate effects otherwise only possible with a volumetric mesh representation.

A further improvement is applied to increase the processing speed. We parallelize computations on heightmaps, i.e. every point of the grid is consider independently from other points on the same grid. It just contributes to computations with corresponding points (same coordinates) on other grids.

Also the conversion from a sketch to a heightmap which requires distance calculations to the line segments on each sketch is performed in parallell on the GPU. Parallelizations through GPU is done every time there is a conversion between relative and absolute layers or vice versa. To generate interactive illustrative models and animations we exploit the efficiency of parallel computation. In case of compaction and faulting, there is only one value that makes the illustration change. In compaction, this value is the rate of compression of the layers; in faulting, it is the vector that defines the displacement of the layers on one side of the fault. In both circumstances, none of the values of the heightmaps changes, they are just either shifted in their position or scaled by a coefficient. For that purpose, it is not necessary to update every heightmap from CPU, but rather go through them in a parallel on the GPU.

C.5 Results

Here we present several examples demonstrating our proposed techniques. Figure C.2 (left side) and Figure C.1 show reverse faults (two blocks pushing against each other) defined by a planar surface and by a non-planar fault respectively. In case of a fault, we can choose whether to have conservation of layer volumes (as in Figure C.6 bottom and Figure C.3 (d)), which is more teaching oriented, or not (as in Figure C.3 (c)), which is a more realistic situation. The former is obtained by setting the bottom layer fully transparent. Compaction is conveyed in the lower layers of Figure C.8; the more deposit, the more compacted is the volume below the deposition. Figure C.11 displays three time steps of a faulting process and, at the same time, reveals the user interface we use to create our models. In Figure C.10, we show different kinds of visualizations of the same

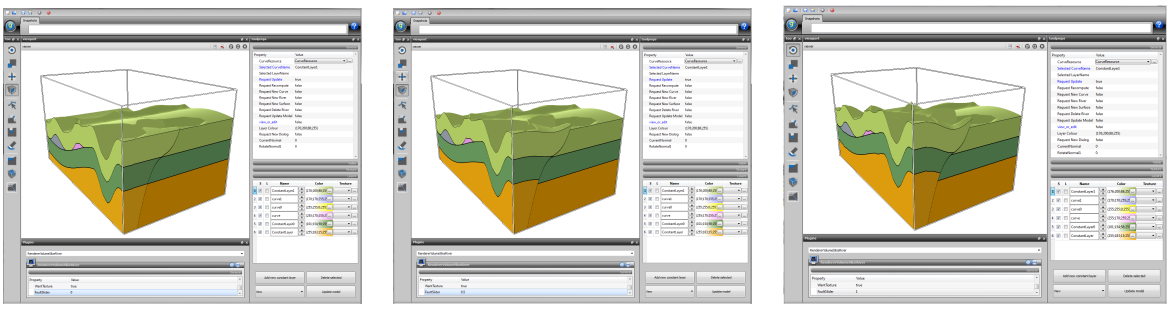


Figure C.11: In this images we reveal the aspect of the user interface, within which an illustration of a faulted model has been built: in the left image the model is faulted but not displaced; in the middle image it is half-way displaced and completely displaced in the image to the right.

model: (a) the entire volume; (b) the volume clipped by a cutting plane; (c) employing a temporal cut; (d) staircase view; (e) exploded view of the layer-cake.

We have placed two illustrations of the same geological model side by side (see Figure C.2) as benchmarking (i.e. comparison with a reference illustration): the one to the left is a 2D image authored by a geologist, the illustration to the right is a 3D volume that has been created using our approach described in this paper.

For our examples, we used a 512×512 grid resolution. In general, each model is independent of metric unit; rock layers and faults can have a large scale range, but, to get an idea of the dimension of the layers a model is trying to reproduce, the reader can think of layer thickness in the order of kilometers.

Table C.1 shows the time that was required to generate some of the examples in the paper.

Model	Time (design and computation)
Figure C.1 (a1-a2)	$<$ 2 minutes
Figure C.1 (b1-b2)	$<$ 2 minutes
Figure C.2 (right model)	$<$ 1 minute
Figure C.3	$<$ 1 minute
Figure C.8	$<$ 2 minutes
Figure C.11	$<$ 1 minute

Table C.1: Some of our models with their respective modelling times.

Techniques that are traditionally used to depict subsurface phenomena would require much more time, as confirmed by geologists who collaborated with us and tried our prototype system. Two-dimensional drawings obtained by hand or software are examples of commonly used methods. Procedural 3D modelling is less widely employed, requires more practice and is not developed for geological illustrations. One of the advantages of our approach is that it permits an interactive process of developing new ideas in a discussion within a group of domain experts. Moreover, a rapid approach to interactively model illustrations of geological phenomena helps to explain ideas to others, as our geologist collaborators pointed out.

C.6 Conclusions

We presented a new approach to rapid model subsurface geology to obtain illustrations and animations. In particular, we focus on creating illustrative visualizations of scenarios showing terrain stratigraphy, internal structures (such as channels), compaction of layers and faulting processes. We provide more than one type of visualization to be able to visually access features inside the volume. Geological events are modelled by means of sketches and real-time interaction is achieved with parallelization of our technique.

Limitations. Employment of heightmaps in the representation sets some limitations related to the shape of geologic features. For instance, a fault that is perfectly vertical could not be handled by our representation. However, most real situations can be achieved with our method. In our current implementation, we are able to achieve real-time interaction with up to about fifty layers (with grid resolution of 512×512).

Future Work. As a future improvement of our technique, we may have to define fault surfaces by meshes instead of heightmaps, so that we do not have to specify a fault surface on a whole grid. This can be useful when dealing with blind faults.

Acknowledgment

The authors would like to thank all reviewers that gave comments and suggestions for improvements, Paolo Angelelli and Ivan Viola for valuable feedback and expertise. This work is funded by the Petromaks program of the Norwegian Research Council through the Geoillustrator project (#200512). This paper has been also supported by the the PhysioIllustration research project (#218023), which is funded by the Norwegian Research Council.

Bibliography

[1] http://www.iris.edu/gifs/animations/faults.htm.

[2] *Computational Geometry Algorithms Library*. http://www.cgal.org.

[3] *SketchUp*. http://sketchup.google.com/.

[4] Abrahamsen, P., Kolbjoernsen, O., and Hauge, R. *Geostatistics Oslo 2012*. Springer, 2012.

[5] Amorim, R., Brazil, E. V., Patel, D., and Sousa, M. C. Sketch modeling of seismic horizons from uncertainty. In *Proceedings Symposium on Sketch-Based Interfaces and Modeling* (2012), pp. 1–10.

[6] Andersen, C., and van Wijngaarden, A.-J. Interpretation of 4d avo inversion results using rock physics templates and virtual reality visualization, north sea examples. *SEG Annual Meeting* (September 2007).

[7] Arpat, G., and Caers, J. A multiple-scale, pattern-based approach to sequential simulation. vol. 14 of *Quantitative Geology and Geostatistics*. 2005, pp. 255–264.

[8] Baojun, W., Bin, S., and Zhen, S. A simple approach to 3D geological modelling and visualization. *Bulletin of Engineering Geology and the Environment 68*, 4 (2009), 559–565.

[9] Belhadj, F. Terrain modeling: a constrained fractal model. In *Proceedings of the 5th international conference on Computer graphics, virtual reality, visualisation and interaction in Africa* (2007), vol. 1, pp. 197–204.

[10] Benes, B., and Forsbach, R. Layered data representation for visual simulation of terrain erosion. *Spring Conference on Computer Graphics (SCCG)* (2001).

[11] Benes, B., and Forsbach, R. Layered data representation for visual simulation of terrain erosion. In *Proceedings of Spring Conference on Computer Graphics* (2001), pp. 80–85.

[12] Benes, B., and Forsbach, R. Visual simulation of hydraulic erosion. *Journal of WSCG* (2002).

[13] Benes, B., Tesinsky, V., Hornys, J., and Bhatia, S. K. Hydraulic erosion. *Computer Animation and Virtual Worlds 17*, 2 (2006), 99–108.

[14] Bernhardt, A., Maximo, A., Velho, L., Hnaidi, H., and Cani, M.-P. Real-time terrain modeling using CPU-GPU coupled computation. In *Conference on Graphics, Patterns and Images, 24th SIBGRAPI* (August 2011), pp. 64–71.

[15] Bertoncello, A., Sun, T., Li, H., Marietho z, G., and Caers, J. Conditioning surface-based geological models to well and thickness data. *Mathematical Geosciences 45*, 7 (2013), 873–893.

[16] Botsch, M., and Kobbelt, L. An intuitive framework for real-time freeform modeling. In *ACM SIGGRAPH 2004 Papers* (2004), SIGGRAPH '04, pp. 630–634.

[17] Bridge, J. The interaction between channel geometry, water flow, sediment transport and deposition in braided rivers. In *Braided Rivers*, vol. 75. Geological Society Special Publication, 1993, pp. 13–71.

[18] Brosz, J., Samavati, F., and Sousa, M. Terrain synthesis by-example. *Advances in Computer Graphics and Computer Vision* (2007), 58–77.

[19] Bruckner, S., and Gröller, M. E. Volumeshop: An interactive system for direct volume illustration. In *Proceedings of IEEE Visualization 2005* (Oct. 2005), H. R. C. T. Silva, E. Gröller, Ed., pp. 671–678.

[20] Bujans, R. A. A thesis on sketch-based techniques for mesh deformation and editing, 2006.

[21] Caers, J. Geostatistical reservoir modelling using statistical pattern recognition. *Journal of Petroleum Science and Engineering 29*, 3 (2001), 177–188.

[22] Caumon, G. Towards stochastic time-varying geological modeling. *Mathematical Geosciences 42*, 5 (2010), 1–25.

[23] Caumon, G., Collon-Drouaillet, P., Le Carlier de Veslud, C., Viseur, S., and Sausse, J. Surface-Based 3D Modeling of Geological Structures. *Mathematical Geosciences 41*, 8 (2009), 927–945.

[24] Caumon, G., Lepage, F., Sword, C. H., and Mallet, J.-L. Building and Editing a Sealed Geological Model. *Mathematical Geology 36*, 4 (2004), 405–424.

[25] Chaikin, G. An algorithm for high speed curve generation. *Computer Graphics and Image Processing 3* (1974), 346–349.

[26] Cherlin, J. J., Samavati, F., Sousa, M. C., and Jorge, J. A. Sketch-based modeling with few strokes. In *Proceedings of SCCG '05* (2005), pp. 137–145.

[27] Chiba, N., and Muraoka, K. An erosion model based on velocity fields for the visual simulation of mountain scenery. *The Journal of Visualization 194* (1998).

[28] Chilès, J.-P., and Delfiner, P. *Geostatistics : modeling spatial uncertainty.* John Wiley and Sons, Inc., 1999.

[29] Cohen, J. M., Hughes, J. F., and Zeleznik, R. C. Harold: a world made of drawings. In *Proceedings of NPAR '00* (2000), pp. 83–90.

[30] Cook, M. T., and Agah, A. A survey of sketch-based 3-d modeling techniques. *Interact. Comput. 21* (July 2009), 201–211.

[31] Correa, C. D., Silver, D., and Chen, M. Illustrative deformation for data exploration. *IEEE Trans. Vis. Comput. Graph. 13*, 6 (2007), 1320–1327.

[32] Coulthard, T. J., and Van De Wiel, M. J. Modelling river history and evolution. *Philosophical Transactions of A Mathematical, Physical and Engineering Sciences 370*, 1966 (2012), 2123–2142.

[33] Cruz, L. M. V., and Velho, L. A sketch on sketch-based interfaces and modeling. *Graphics, Patterns and Images Tutorials, SIBGRAPI* (2010), 22–33.

[34] Cutler, B., Dorsey, J., McMillan, L., Müller, M., and Jagnow, R. A procedural approach to authoring solid models. *ACM Transactions on Graphics 21*, 3 (July 2002), 302–311.

[35] de Araújo, B. R., Redol, R. A., Armando, J., and Jorge, P. Blobmaker: Free-form modelling with variational implicit surfaces. In *In Proc. of the 12th Portuguese Computer Graphics Meeting* (2003), pp. 17–26.

[36] de Carpentier, G. J. P., and Bidarra, R. Interactive gpu-based procedural heightfield brushes. In *Proceedings of the 4th International Conference on Foundations of Digital Games* (New York, NY, USA, 2009), FDG '09, ACM, pp. 55–62.

[37] De Kemp, E., and Sprague, K. Interpretive tools for 3-D structural geological modeling part I: Bezier-based curves, ribbons and grip frames. *Geoinformatica 7*, 1 (2003), 55–71.

[38] Desbrun, M., Meyer, M., and Alliez, P. Intrinsic parameterizations of surface meshes. *Comput. Graph. Forum 21*, 3 (2002), 209–218.

[39] Dhont, D., Luxey, P., and Chorowicz, J. 3-D modeling of geologic maps from surface data. *AAPG Bulletin 89*, 11 (2005), 1465–1474.

[40] Doran, J., and Parberry, I. Controlled procedural terrain generation using software agents. *IEEE Transactions on Computational Intelligence and AI in Games 2*, 2 (2010), 111–119.

[41] Dorsey, J., Edelman, A., Jensen, H., Legakis, J., and Pedersen, H. Modeling and rendering of weathered stone. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques* (1999), pp. 225–234.

[42] Eggli, L., Hsu, C., Brüderlin, B. D., and Elber, G. Inferring 3D models from freehand sketches and constraints. *Computer-Aided Design 29*, 2 (1997), 101 – 112.

[43] feng Zhu, L., He, Z., Pan, X., and cai Wu, X. An approach to computer modeling of geological faults in 3d and an application. *Journal of China University of Mining and Technology 16*, 4 (2006), 461 – 465.

[44] Floater, M. S., and Hormann, K. Surface parameterization: a tutorial and survey. In *Advances in Multiresolution for Geometric Modelling*. 2005, pp. 157–186.

[45] Floater, Michael and Halbwachs, Yvon and Hjelle, Øyvind and Reimers,Martin. Omega: A cad-based approach to geological modelling. GOCAD ENSG, Conference Proceedings.

[46] Fossen, H. *Geologi. Stein, mineraler, fossiler og olje.* Fagbokforlaget, 2008.

[47] Frank, T. Geological information retrieval using tetrahedral meshes. *gocad.org* (2006), 12–15.

[48] Gain, J., Marais, P., and Strasser, W. Terrain sketching. In *Proceedings of Symposium on Interactive 3D Graphics and Games* (2009), pp. 31–38.

[49] Gamito, M. N., and Maddock, S. C. Topological correction of hypertextured implicit surfaces for ray casting. *The Visual Computer 24* (2008), 397–409.

[50] Gani, M. R., and Bhattacharya, J. P. Basic building blocks and process variability of a cretaceous delta: Internal facies architecture reveals a more dynamic interaction of river, wave, and tidal processes than is indicated by external shape. *Journal of Sedimentary Research 77* (2007), 284–302.

[51] Gingold, R. A., and Monaghan, J. J. Smoothed particle hydrodynamics: Theory and application to non-spherical stars. *Monthly Notices of the Royal Astronomical Society 181* (1977), 375.

[52] Groshong, R. *3-D Structural Geology: A Practical Guide to Surface and Subsurface Map Interpretation.* Springer, 1999.

[53] Grundvåg, S.-A. *Outcrop and subsurface characterization of shelf-margin clinoform systems.* PhD thesis, University of Bergen, 2012.

[54] Guillen, a., Calcagno, P., Courrioux, G., Joly, A., and Ledru, P. Geological modelling from field data and geological knowledge Part II. Modelling validation using gravity and magnetic data inversion. *Physics of the Earth and Planetary Interiors 171*, 1-4 (2008), 158–169.

[55] Halbwachs, Yvon and Hjelle, Øyvind. *Generalized maps in geological modeling: Object-oriented design of topological kernels.* Advances in Software Tools for Scientific Computing. Springer Verlag. 2002.

[56] Hinderer, M. From gullies to mountain belts: A review of sediment budgets at various scales. *Sedimentary Geology 280* (2012), 21–59.

[57] Hnaidi, H., Guérin, E., Akkouche, S., Peytavie, A., and Galin, E. Feature based terrain generation using diffusion equation. *Computer Graphics Forum 29*, 7 (2010), 2179–2186.

[58] Houlding, S. *3D Geoscience Modeling: Computer Techniques for Geological Characterization.* Springer, 1994.

[59] Hudak, M., and Durikovic, R. Terrain Models for Mass Movement Erosion. In *Proceedings of EG-UK TPCG Conference* (2011).

[60] Igarashi, T., Matsuoka, S., and Tanaka, H. Teddy: a sketching interface for 3D freeform design. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques* (New York, NY, USA, 1999), SIGGRAPH '99, ACM Press/Addison-Wesley Publishing Co., pp. 409–416.

[61] Joshi, P. Curve-based shape modeling - a tutorial. *IEEE Comput. Graph. Appl. 31* (Nov. 2011), 18–23.

[62] Kadlec, B. J., Tufo, H. M., and Dorn, G. A. Knowledge-assisted visualization and segmentation of geologic features. *IEEE Computer Graphics and Applications 30*, 1 (Jan. 2010), 30–39.

[63] Karpenko, O. A., and Hughes, J. F. Smoothsketch: 3d free-form shapes from complex sketches. *ACM Trans. Graph. 25*, 3 (July 2006), 589–598.

[64] Kartasheva, E., Adzhiev, V., Comninos, P., Fryazinov, O., and Pasko, A. Heterogeneous objects modelling and applications. Springer-Verlag, Berlin, Heidelberg, 2008, ch. An implicit complexes framework for heterogeneous objects modelling, pp. 1–41.

[65] Kaufmann, O., and Martin, T. 3D geological modelling from boreholes, cross-sections and geological maps, application over former natural gas storages in coal mines. *Computers & Geosciences 34*, 3 (2008), 278–290.

[66] Klausen, T., Helland-Hansen, W., Laursen, I., and Gawthorpe, R. Fluvial geometries and reservoir characteristics of a triassic coastal plain system in the norwegian barents sea. *AAPG Search and Discovery* (2012).

[67] Klausen, T. G. *Sedimentology, reservoir geometry and paleogeography of the Mid- to Late Triassic Snadd Formation of the Barents Sea*. PhD thesis, University of Bergen, 2013.

[68] Krištof, P., Beneš, B., Křivánek, J., and Šťava, O. Hydraulic Erosion Using Smoothed Particle Hydrodynamics. *Computer Graphics Forum 28*, 2 (2009), 219–228.

[69] Lemon, A., and Jones, N. Building solid models from boreholes and user-defined cross-sections. *Computers & Geosciences 29*, 5 (2003), 547–555.

[70] Lévy, B., Petitjean, S., Ray, N., and Maillot, J. Least squares conformal maps for automatic texture atlas generation. In *Proceedings of SIGGRAPH '02* (2002), pp. 362–371.

[71] Lidal, E. M., Hauser, H., and Viola, I. Geological storytelling - graphically exploring and communicating geological sketches. In *Proceedings Symposium on Sketch-Based Interfaces and Modeling* (2012), pp. 11–20.

[72] Lidal, E. M., Langeland, T., Giertsen, C., Grimsgaard, J., and Helland, R. A decade of increased oil recovery in virtual reality. *IEEE Computer Graphics and Applications 27*, 6 (2007), 94–97.

[73] Lidal, E. M., Natali, M., Patel, D., Hauser, H., and Viola, I. Geological storytelling. *Computers & Graphics 37* (2013), 445–459.

[74] Lopez, S., Galli, A., and Cojan, I. Fluvial meandering channelized reservoirs: a stochastic and process-based approach. In *Proceedings Annual Conference of the International Association of Mathematical Geologists* (2001).

[75] Lucy, L. A numerical approach to the testing of the fission hypothesis. *Astronomical Journal 82* (1977), 1013.

[76] Mallet, J. Discrete modeling for natural objects. *Mathematical geology 29*, 2 (1997), 199–219.

[77] Mallet, J.-L. Discrete smooth interpolation. *ACM Transactions on Graphics 8*, 2 (Apr. 1989), 121–144.

[78] Mallet, J.-l. Discrete smooth interpolation in geometric modelling. *Computer-aided design 24*, 4 (1992), 178–191.

[79] Mandelbrot, B. B. *The Fractal Geometry of Nature*. W. H. Freedman and Co., New York, 1982.

[80] Marroquim, R., Cavalcanti, P. R., and Esperança, C. 3d adaptive multiresolution triangulations based on physical compression. In *Proceedings of the XXV Iberian Latin American Congress on Computational Methods (CILAMCE), Recife-PE* (November 2004).

[81] Masry, M., and Lipson, H. A sketch-based interface for iterative design and analysis of 3d objects. In *ACM SIGGRAPH 2007 courses* (New York, NY, USA, 2007), SIGGRAPH '07, ACM.

[82] McInerney, P., Goldberg, A., Calcagno, P., Courrioux, G., Guillen, A., and Seikel, R. Improved 3D geology modelling using an implicit function interpolator and forward modelling of potential field data. In *Proceedings of Exploration* (2007), vol. 7, pp. 919–922.

[83] Michaels, H., Gorelick, S., Sun, T. Li, H., Boucher, A., and Caers, J. Combining geologic-process models and geostatistics for conditional simulation of 3-d subsurface heterogeneity. *Water Resources Research 46* (2010).

[84] Ming, J., and Pan, M. An Improved Horizons Method for 3D Geological Modeling from Boreholes. *2009 International Conference on Environmental Science and Information Application Technology* (2009), 369–374.

[85] Mori, Y., and Igarashi, T. Plushie: an interactive design system for plush toys. In *Proceedings of SIGGRAPH '07* (2007), pp. 1–8.

[86] Mullen, P., Tong, Y., Alliez, P., and Desbrun, M. Spectral conformal parameterization. *Comput. Graph. Forum 27*, 5 (2008), 1487–1494.

[87] Musgrave, F., Kolb, C., and Mace, R. The synthesis and rendering of eroded fractal terrains. *ACM SIGGRAPH Computer Graphics 23*, 3 (1989), 41–50.

[88] Natali, M. Illustrative modelling of fluvial systems. `folk.uib.no/mna024/public/tutorial.mp4`, 2013.

[89] Natali, M., Klausen, T. G., and Patel, D. Sketch-based modelling and visualization of geological deposition. *Computers & Geosciences 67C* (2014), 40–48.

[90] Natali, M., Lidal, E. M., Parulek, J., Viola, I., and Patel, D. Modeling terrains and subsurface geology. In *Proceedings of EuroGraphics 2013 (STARs)* (Girona, Spain, 2013).

[91] Natali, M., Viola, I., and Patel, D. Rapid visualization of geological concepts. In *Proceedings 25th SIBGRAPI Conference on Graphics, Patterns and Images* (2012).

[92] Neidhold, B., Wacker, M., and Deussen, O. Interactive physically based fluid and erosion simulation. In *Proceedings of the First Eurographics Conference on Natural Phenomena* (2005), NPH '05, Eurographics Association, pp. 25–33.

[93] Olsen, J. Realtime procedural terrain generation. Tech. rep., Department of Mathematics And Computer Science (IMADA) University of Southern Denmark, 2004.

[94] Olsen, L., Samavati, F., and Jorge, J. Naturasketch: Modeling from images and natural sketches. *IEEE Computer Graphics and Applications 31*, 6 (Nov. 2011), 24–34.

[95] Olsen, L., Samavati, F., Sousa, M., and Jorge, J. Sketch-based modeling: A survey. *Computers & Graphics 33* (2009), 85–103.

[96] Olsen, L., and Samavati, F. F. Image-assisted modeling from sketches. In *Proceedings of Graphics Interface 2010* (Toronto, Ont., Canada, Canada, 2010), GI '10, Canadian Information Processing Society, pp. 225–232.

[97] Orzan, A., Bousseau, A., Winnemöller, H., Barla, P., Thollot, J., and Salesin, D. Diffusion curves: a vector representation for smooth-shaded images. *ACM Transactions on Graphics 27*, 3 (Aug. 2008).

[98] Owada, S., Nielsen, F., Okabe, M., and Igarashi, T. Volumetric illustration: Designing 3d models with internal textures. *ACM Trans. Graph. 23*, 3 (Aug. 2004), 322–328.

[99] Parks, D. Freeform modeling of faulted surfaces in seismic images. *SEG Technical Program Expanded Abstracts 28*, 1 (2009), 2702–2706.

[100] Pasko, A. A., Adzhiev, V., Sourin, A., and Savchenko, V. V. Function representation in geometric modeling: concepts, implementation and applications. *The Visual Computer 11*, 8 (1995), 429–446.

[101] Patel, D., Bruckner, S., Viola, I., and Gröller, M. E. Seismic Volume Visualization for Horizon Extraction. In *IEEE Pacific Visualization 2010* (2010), pp. 73–80.

[102] Patel, D., Giertsen, C., Thurmond, J., Gjelberg, J., and Gröller, M. E. The seismic analyzer: interpreting and illustrating 2D seismic data. *IEEE transactions on visualization and computer graphics 14*, 6 (2008), 1571–8.

[103] Patel, D., Giertsen, C., Thurmond, J., and Gröller, M. Illustrative rendering of seismic data. In *Proceedings of vision modeling and visualization* (2007), pp. 13–22.

[104] Patel, D., Giertsen, C., Thurmond, J., and Gröller, M. Illustrative rendering of seismic data. In *Proceedings of vision modeling and visualization* (2007), pp. 13–22.

[105] Peytavie, a., Galin, E., Grosjean, J., and Merillou, S. Arches: a Framework for Modeling Complex Terrains. *Computer Graphics Forum 28*, 2 (2009), 457–467.

[106] Peytavie, a., Galin, E., Grosjean, J., and Merillou, S. Procedural Generation of Rock Piles using Aperiodic Tiling. *Computer Graphics Forum 28*, 7 (2009), 1801–1809.

[107] Peytavie, A., Galin, E., Merillou, S., and Grosjean, J. Arches: a Framework for Modeling Complex Terrains. *Computer Graphics Forum (Proceedings of Eurographics) 28*, 2 (2009), 457–467.

[108] Pinkall, U., and Polthier, K. Computing discrete minimal surfaces and their conjugates. *Experimental Mathematics 2* (1993), 15–36.

[109] Plate, J., Tirtasana, M., Carmona, R., and Fröhlich, B. Octreemizer: a hierarchical approach for interactive roaming through very large volumes. In *Proceedings of the Symposium on Data Visualisation* (2002), VISSYM '02, Eurographics Association, pp. 53–ff.

[110] Porębski, S. J., and Steel, R. J. Shelf-margin deltas: their stratigraphic significance and relation to deepwater sands. *Earth-Science Reviews 62* (2003), 283–326.

[111] Preparata, F. P., and Shamos, M. I. *Computational geometry: an introduction*. Springer-Verlag New York, Inc., New York, NY, USA, 1985.

[112] Pringle, J., Howell, J., Hodgetts, D., Westerman, A., and Hodgson, D. Virtual outcrop models of petroleum reservoir analogues: a review of the current state-of-the-art. *first break 24* (2006), 33.

[113] Prusinkiewicz, P., Hammel, M., and Tn, C. A fractal model of mountains with rivers. *Panorama* (1993), 174–180.

[114] Pyrcz, M. *The integration of geologic information into geostatistical models*. PhD thesis, University of Alberta, Edmonton, Alberta, 2004.

[115] Pyrcz, M., Catuneanu, O., and Deutsch, C. Stochastic surface-based modeling of turbidite lobes. *American Association of Petroleum Geologists Bulletin 89*, 2 (2005), 177–191.

[116] Pyrcz, M., McHargue, T., Clark, J., Sullivan, M., and Strebelle, S. Event-based geostatistical modeling: Description and applications. In *Geostatistical Congress, Oslo, Norway* (2012).

[117] Reading, H. *Sedimentary Environments: Processes, Facies and Stratigraphy.* John Wiley & Sons, 1996.

[118] Rivers, A., Durand, F., and Igarashi, T. 3d modeling with silhouettes. *ACM Trans. Graph. 29*, 4 (July 2010), 109:1–109:8.

[119] Roudier, P., Peroche, B., and Perrin, M. Landscapes synthesis achieved through erosion and deposition process simulation. *Computer Graphics Forum 12*, 3 (1993), 375–383.

[120] Sapozhnikov, V., and Nikora, V. Simple computer model of a fractal river network with fractal individual watercourses. *Journal of Physics A: Mathematical and General 26* (1993), L623.

[121] Schneider, J., Boldte, T., and Westermann, R. Real-time editing, synthesis, and rendering of infinite landscapes on GPUs. In *Vision, modeling, and visualization 2006: proceedings, November 22-24, 2006, Aachen, Germany* (2006), p. 145.

[122] Sibson, R. A brief description of natural neighbor interpolation. *Interpreting mulitvariate data* (1981), 21–26.

[123] SIS. Petrel seismic interpretation software. *Schlumberger Information Solutions*. `http://www.slb.com/services/software/geo/petrel/seismic/seismic_interpretation.aspx`.

[124] Sousa, M. C., Ebert, D. S., Stredney, D., and Svakhine, N. A. Illustrative visualization for medical training. In *Computational Aesthetics* (2005), pp. 201–208.

[125] Sprague, K. B., and de Kemp, E. a. Interpretive Tools for 3-D Structural Geological Modelling Part II: Surface Design from Sparse Spatial Data. *GeoInformatica 9*, 1 (2005), 5–32.

[126] Stachniak, S., and Stuerzlinger, W. An algorithm for automated fractal terrain deformation. *Computer Graphics and Artificial Intelligence* (2005).

[127] Stava, O., Benes, B., Brisbin, M., and Krivanek, J. Interactive terrain modeling using hydraulic erosion. In *Proceedings ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2008), pp. 201–210.

[128] Strebelle, S. Conditional Simulation of Complex Geological Structures Using Multiple-Point Statistics. *Mathematical Geology 34*, 1 (2002).

[129] Takayama, K., Okabe, M., Ijiri, T., and Igarashi, T. Lapped solid textures: filling a model with anisotropic textures. *ACM Transactions on Graphics (proceedings of ACM SIGGRAPH) 27*, 3 (2008), 1–9.

[130] Takayama, K., Sorkine, O., Nealen, A., and Igarashi, T. Volumetric modeling with diffusion surfaces. *ACM Transactions on Graphics (proceedings of ACM SIGGRAPH) 29*, 6 (2010).

[131] Thompson, G., and Turk, J. *Introduction to physical geology*. Saunders Golden Sunburst Series. Saunders College Pub., 1998.

[132] Turner, A., and Gable, C. A review of geological modeling. *Three-dimensional geologic mapping for groundwater applications. Minnesota Geological Survey Open-file Report* (2007), 07–4.

[133] Turner, a. K. Challenges and trends for geological modelling and visualisation. *Bulletin of Engineering Geology and the Environment 65*, 2 (2006), 109–127.

[134] Viard, T., Caumon, G., and Lévy, B. Adjacent versus coincident representations of geospatial uncertainty: Which promote better decisions? *Computers & Geosciences* (2010).

[135] Viola, I. *Using* SketchUp *for creating a layer-cake model*, 2011. `http://dl.dropbox.com/u/44934338/layercake_screencapture.wmv`.

[136] Viola, I. Using *SketchUp* for editing a layer-cake model and producing a fault. `http://dl.dropbox.com/u/44934338/fault_screencapture.wmv`.

[137] Viola, I., Sousa, M. C., Ebert, D., Preim, B., Gooch, B., Andrews, B., and Tietjen, C. Illustrative visualization for science and medicine. In *Eurographics Tutorial* (2006).

[138] Vital Brazil, E., Macêdo, I., costa Sousa, M., de Figueiredo, L. H., and Velho, L. Sketching Variational Hermite-RBF Implicits. In *Proc. Sketch Based Interfaces and Modeling* (2010), pp. 1–8.

[139] Vital Brazil, E., Macedo, I., Costa Sousa, M., de Figueiredo, L. H., and Velho, L. Sketching variational Hermite-RBF implicits. In *Proceedings of SBIM '10* (2010), pp. 1–8.

[140] Wang, L., Yu, Y., Zhou, K., and Guo, B. Multiscale vector volumes. *ACM Transactions on Graphics 30*, 6 (Dec. 2011), 167:1–167:8.

[141] Watanabe, N., and Igarashi, T. A sketching interface for terrain modeling. In *SIGGRAPH '04 Posters* (2004), pp. 73–73.

[142] Wijns, C. Inverse modelling in geology by interactive evolutionary computation. *Journal of Structural Geology 25*, 10 (2003), 1615–1621.

[143] Wu, Q, X. H. An approach to computer modeling and visualization of geological faults in 3D. *Computers & Geosciences 29*, 4 (2003), 503–509.

[144] Xie, Y., Deutsch, C. V., and Cullick, A. S. Surface-geometry and trend modeling for integration of stratigraphic data in reservoir models. In *GEOSTATS 2000, Geostatistical Congress* (April 2000).

[145] Zhang, G.-X., Du, S.-P., Lai, Y.-K., Ni, T., and Hu, S.-M. Sketch guided solid texturing. *Graphical Models 73*, 3 (2011), 59–73.

[146] Zhang, Z. 3D Terrain Reconstruction Based on Contours. *Computer* (2005), 3–8.

[147] Zhou, H., Sun, J., Turk, G., and Rehg, J. M. Terrain synthesis from digital elevation models. *IEEE Transactions on Visualization and Computer Graphics 13*, 4 (July/August 2007), 834–848.

[148] Zhu, B., Iwata, M., Haraguchi, R., Ashihara, T., Umetani, N., Igarashi, T., and Nakazawa, K. Sketch-based dynamic illustration of fluid systems. In *Proceedings SIGGRAPH Asia* (2011), SA '11, pp. 134:1–134:8.

[149] Zhu, B., Iwata, M., Haraguchi, R., Ashihara, T., Umetani, N., Igarashi, T., and Nakazawa, K. Sketch-based dynamic illustration of fluid systems. *ACM Transaction on Graphics 30*, 6 (Dec. 2011), 134:1–134:8.