REPORTS
IN
INFORMATICS

ISSN 0333-3590

Sketch-Based Modeling and Visualization of Geological Deposition

Mattia Natali, Tore Grane Klausen, Daniel Patel

REPORT NO 406 **September 2013**

Department of Informatics

UNIVERSITY OF BERGEN

Bergen, Norway

This report has URL
http://www.ii.uib.no/publikasjoner/texrap/pdf/2013-406.pdf

Reports in Informatics from Department of Informatics, University of Bergen, Norway, is available at http://www.ii.uib.no/publikasjoner/texrap/.

Sketch-Based Modeling and Visualization of Geological Deposition

Mattia Natali*

Department of Informatics
University of Bergen
N-5020 Bergen
Norway

Tore Grane Klausen

Department of Earth Science
University of Bergen
N-5020 Bergen
Norway

Daniel Patel

Christian Michelsen Research AS
N-5892 Bergen
Norway

September 20, 2013

Abstract

We propose a method for modelling and visualizing geological scenarios by sequentially disposing stratigraphy layers. Evolution of rivers and deltas is important for geologists when interpreting the stratigraphy of the subsurface, in particular for hydrocarbon exploration. We illustratively visualize 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. We use a data structure consisting of a stack of layers, where each layer represents a unique erosion or deposition event.

1 Introduction

Geologists are interested in improving their tools for modelling and visualizing their ideas on earth behaviour. They want to do this in an expressive and simple way, which is particularly important for communicative purposes. Current tools in geology have a high learning curve and are tedious to use. 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 layers of rocks (stratigraphy of the terrain), deltas and rivers. We mainly focus on illustrative visualization and modelling of rivers and deltas, and how they change the morphology of a terrain during their evolution. The history of rivers and deltas is left as an imprint in layers generated by erosion and deposition processes during changes of shape in geological time. Our technique provides a way to sketch and visualize such layers related to the evolution of rivers and deltas (as shown in Figure 1), an important process for geologists who interpret the stratigraphy of the subsurface. Our concise representation is well suited to obtain interactive layer-cake visualizations, resulting in geological illustrations that are helpful for education and communication. A handmade geological illustration of the depositional history of a river is shown in Figure 2 left (illustration drawn by the geologist co-authoring this article). Geological illustrations can contribute to the process of

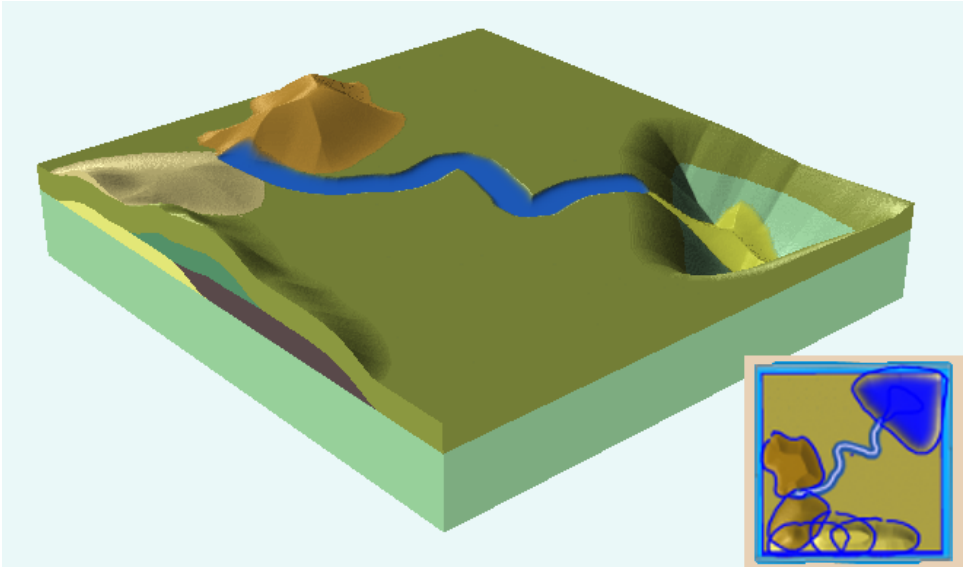

Figure 1: A 3D model created using our sketch-based approach to shape surface and subsurface geological features. Bottom right inset shows a map view of the model together with the sketched strokes.

detecting subsurface resources such as oil, gas, water, minerals and metals. 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 have a high preservation potential (Hinderer [Hin12]). 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 for.

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, by 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 and interpretation (Reading [Rea96]). 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 [GB07], Porębski and Steel [PS03]). In Figure 2 left, an example of a hand-drawn block diagram depicting a meandering river channel is shown. It illustrates the aerial and 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 for multiple cross-section visualizations. This helps in understanding complex internal layering within the sandstone, otherwise not intuitively apprehensible (Bridge [Bri93]). Among the depositional systems that may result in hydrocarbon

*Email: mattia.natali@uib.no

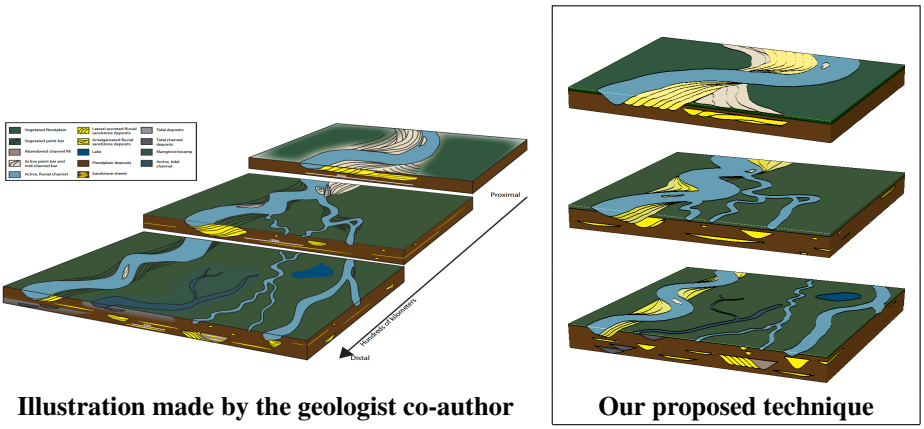

Illustration made by the geologist co-author **Our proposed technique**

Figure 2: Comparison between a 2D illustration of a river sedimentation process and the result achieved with our approach. This example is a real case analysis, where modellization comes from field observations.

accumulation, rivers and deltas are central.

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. [BTHB06] and Stava et al. [SBBK08]). 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 ([HGA+10, GMSe09]), but they only consider the top surface and not subsurface structures, as we do. Some works in geology focus on producing layered representations of the earth by modelling the mesh of each separating surface individually, like Baojun et al. [BBZ09], but is not sketch-based and therefore time consuming. Amorim et al. [ABPS12] address 3D seismic interpretation by modelling one surface at a time, while Cutler et al. [CDM+02] propose a procedural method to obtain layered solid models. None of these two last works consider fluvial systems consisting of lakes, rivers or deltas.

Illustrative Geology For landscape generation and structural geology representations, much of the work presented in literature focuses on realistic rendering and complex procedural and simulation techniques to obtain a model. We are not aware of any work for fast creation of 3D illustrations with internal structures. Geological illustrations let geologists express and explain processes involved in the subsurface. This process of externalization of ideas is a crucial aspect in many fields, including geology (as highlighted by Lidal et al. [LNP+13]). In addition to enabling communication between experts, geological illustrations are heavily employed for teaching purposes. That is because they are easy to understand and expressive at the same time. Sometimes it is not even necessary to support them with an oral explanation, if they are completed with self-descriptive notations. Illustrations in geology are helpful both to show and explain how the earth has evolved and to predict possible future scenarios. A classical usage of these illustrations is for text-book images. We aim to produce images of comparable quality using less time than an illustrator, while supporting interactive exploration since our digital representations are 3D models with internal structures, as opposed to static 2D illustrations made with 2D vector drawing software.

Fluvial Landscape Evolution When the top surface of a geological model contains rivers and streams, or has been shaped by them, it is called 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 [CVDW12]. Models concerning fluvial geomorphology are made for attempting to retrieve the history or to predict the evolution of rivers (one example of illustration in the context of fluvial geomorphology is shown in Figure 2 left). 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 physical-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 ([BF01]), 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 [LJ03], 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, 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 not allowed. Peytavie et al. [PGMG09] 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 3) which can be rendered volumetrically and with arbitrary cuts.

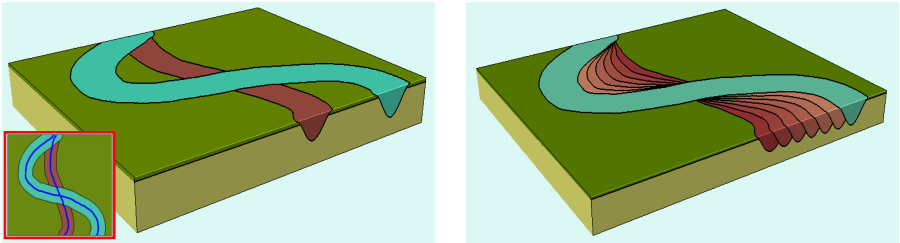

Figure 3: River evolution example. Left: first and last configuration of the river are sketched and imprinted. Right: imprint of additional five intermediate stages of the depositional history.

3 Description of our Approach

The entire approach is based on two data structures; the *absolute* layers and the *relative* layers. The latter lets us keep each geological process independent to all the others with respect to time, that means we can easily rearrange layers in any order without further computations. The former is used in the rendering phase. We now present a formal definition of the two related representations.

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 let us express the i-th absolute layer as

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

Figure 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 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.

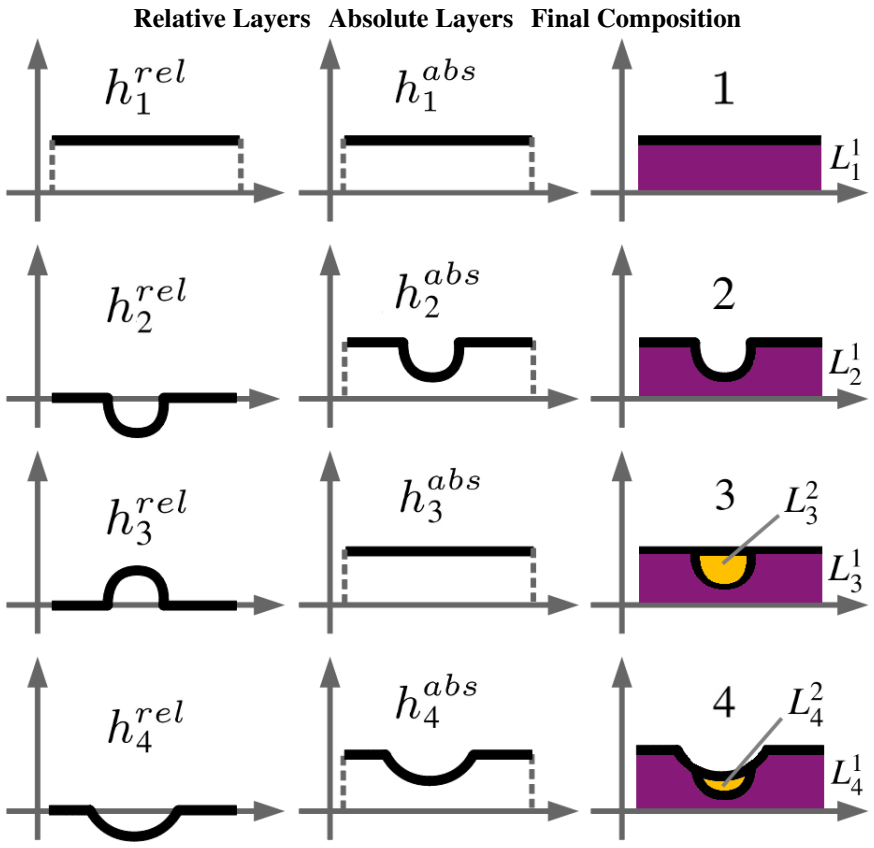

Figure 4: Relation between relative layer, left column and absolute layer, middle column. The right column shows the accumulated final model.

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 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 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.

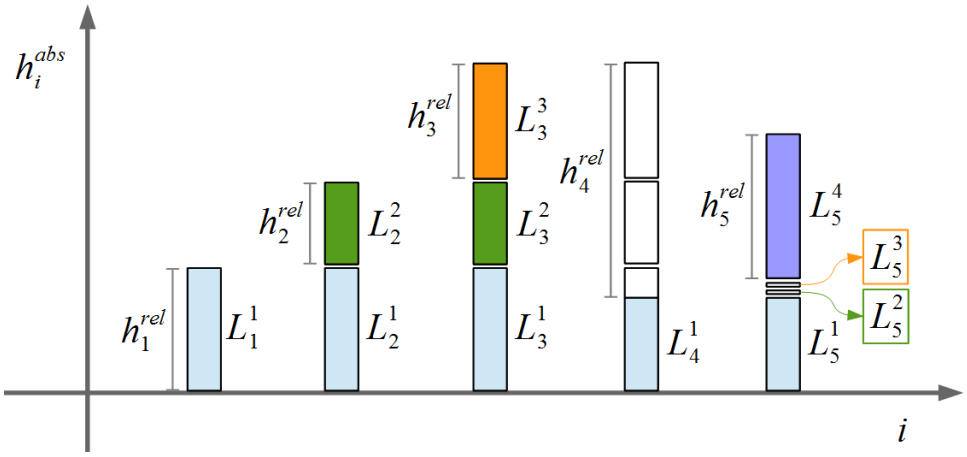

Figure 5: Interaction between the i-th relative layer h_i^{rel} and the previous layers $L_i^1, \ldots, L_i^l$. This example refers to a single point $(\hat{x}, \hat{y})$ of the grid $\mathcal{G}$. Only $h_4^{rel}(\hat{x}, \hat{y}) < 0$, therefore from step 3 to step 4 we observe an erosion.

Our model is obtained through an accumulation of relative layers, defined on uniformly sampled grids represented as 2D height maps. Intuitively, 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 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 4). This relative layer contains negative values that indicate the erosion depth for each point of the grid.

3.2 Rendering the Model

To construct a solid representation of our model handling translucency and arbitrary spatial variations, we render layers using volume rendering with ray-casting. A boundary representation of the solid would require computations of intersections between surfaces and cutting planes and would not directly support volumetric transparency and color variations. Volumetric rendering inherently supports volumetric variations. 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 modelled, or varying layer properties such as grain size in a delta deposit.

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 on the ray,

we identify in which layer we are. The colour and opacity of the sample is given by a user defined mapping from the layer id to an RGBA value. We perform the ray casting on the GPU to enable 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 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 have to quickly identify in which layer a sample at position (x, y, z) is. A naïve approach would sum up the thicknesses of each layer at position (x, y), starting at the bottom layer (index 0), until layer l, where the accumulated value is higher than z. The sample will then be inside the previous layer with index $(l - 1)$. This approach has complexity $O(n)$ in terms of the number of texture lookups that must be performed for each sample. Texture lookups are expensive. To reduce them, we perform a preprocessing step on the relative layers to create the absolute layer structure as described earlier, and use this instead. For each (x, y) position in a layer, we store four components: the starting and ending position of the layer and the index of the next and previous nonempty layer. See Figure 6 and the accompanying Table 1 describing the four values for each of the layers for position $x = x_3$. In Figure 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 $x = x_2$. The key to accelerating the lookups is that we first look up in the same layer as the previous sample belonged to (layer 2), and retrieve the start and 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 1. The layer interval for L2 is below the z value for the sample, so we must search in layers above to find the correct interval. However, several layers (L3, L4, L5) have zero thickness at this x position. During the creation of the data structure, this was identified and a pointer to the next nonempty layer was made. We look up in L6 and check if the z value is within this interval. With reasonably small ray increments, this procedure has complexity $O(1)$.

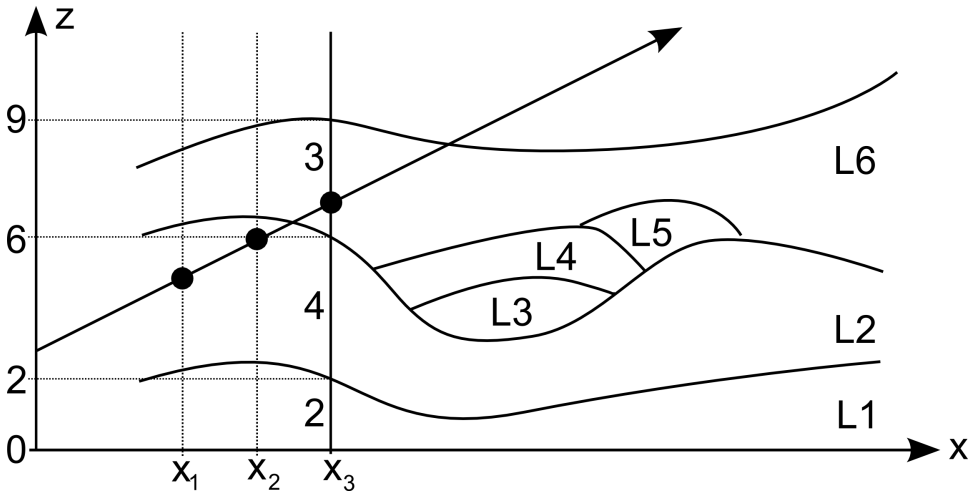

Figure 6: Using the absolute data structure for fast access along a ray cast from left to right.

Global water level is rendered by setting the sample colour to blue and opacity to a user defined water opacity if the z value (height) of the sample is lower than the user defined water level. Water is not rendered in positions where the evaluated layer colour is not fully transparent. Ray color accumulation results in the effect of volumetric attenuation inside translucent water. In the case that all samples in each layer would have the same colour and

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 1: In this table the absolute layer properties for position x_3 of the example in Figure 6 are listed.

opacity, one could perform a dramatically faster rendering by rasterizing the geometry of each layer into a Layered Depth Image ([SGHS98]). Then one would, for each fragment, know the starting and ending position of each layer and could calculate the accumulated colour contribution for each ray-layer intersection analytically. However, 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 7).

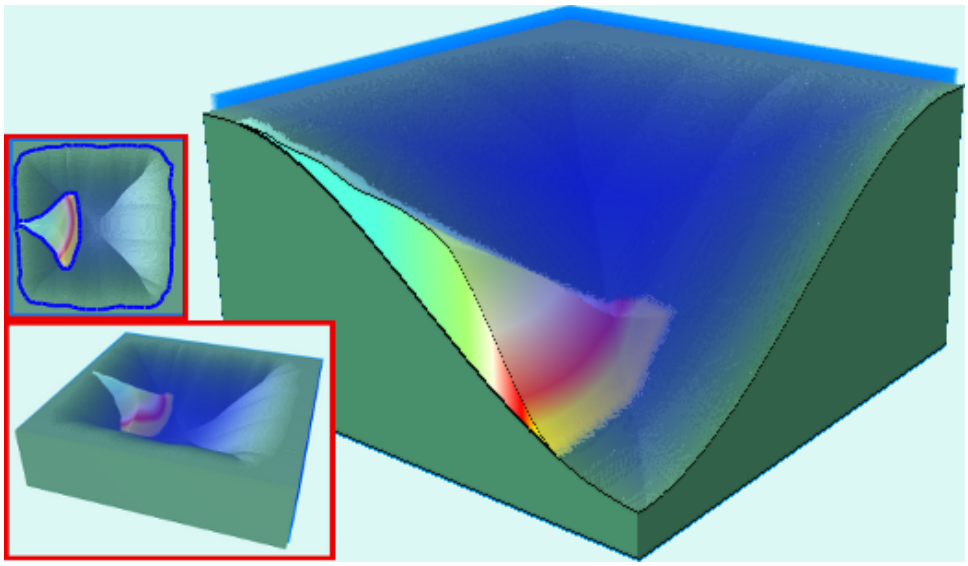

Figure 7: Cross-sectional view showing the internal structure of a delta deposit, where different grain size is conveyed by colour transition. Sea-level is set to be highly translucent.

On cutting planes, a black separator line is drawn between adjacent layers. This makes it possible to distinguish different depositions. However it is possible to assign different layers into the same layer group for avoiding separator lines between them. This is necessary for giving the brown soil in Figure 2 one homogeneous appearance even though it consists of several smaller layers intermixed with river layers.

3.3 Geological Concepts Through Sketches

Our definition of geological features that appear on the model is based on two important metaphors: the river and the delta. These two metaphors are a basis for the definition of all the other operators that generate 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. All the elements composing the layer-cake model are defined by user sketches. An array of sketches is stored, where each sketch is an array of points defining an open or closed curve. Every sketch is subject to a Chaikin subsampling ([Cha74]) to reduce the number of samples while still maintaining its shape.

The main aspect of the model generation and the novelty introduced by our paper is that different stages of the evolution of a river or delta can be defined with few strokes. The depositional history of rivers is defined by sketching only the start and end configuration. Intermediate steps are then calculated as linear interpolations of the first and 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. Given two rivers, defined by two curves C_s and C_e, that respectively describe the starting and 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 $t \in [0, 1]$.

In the following paragraphs, we describe how the sketches are interpreted, classified and converted to relative layers.

Modelling a River When the first and last point of a curve are not close (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 both are river centrelines and close to each other. 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 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. We assume that eroding layers associated with rivers are always coupled with complementary depositional layers, i.e. defined by grids with opposite values. To define intermediate rivers, two rivers are sketched and the number of intermediate rivers is specified.

Figure 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 $L = \sqrt{2}T, T = \frac{||A-B||_2}{2}$.

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 8 bottom).

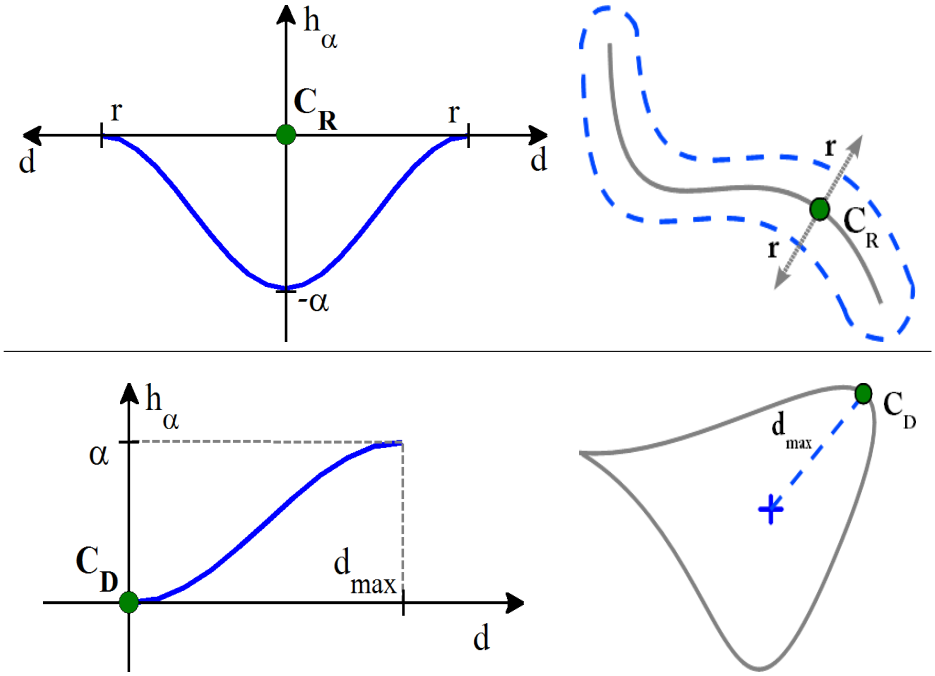

Figure 8: Top: the function h_α representing a river section. C_R is a arbitrary point along the river centreline. Bottom: the function h_α defining delta deposition. C_D is a arbitrary point along the delta contour.

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, differently 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

$$L_i = \max_{W \in C} ||V_i - W||_2.$$

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. The user can also specify if an additional depositon, of 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 9), or mountains with a larger amount of deposition (as in Figure 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 2 right. In addition, layers of constant, user defined thickness may be introduced in the layer-cake by simply drawing a single point.

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 9, where the shape of each lake is defined by a single user stroke.

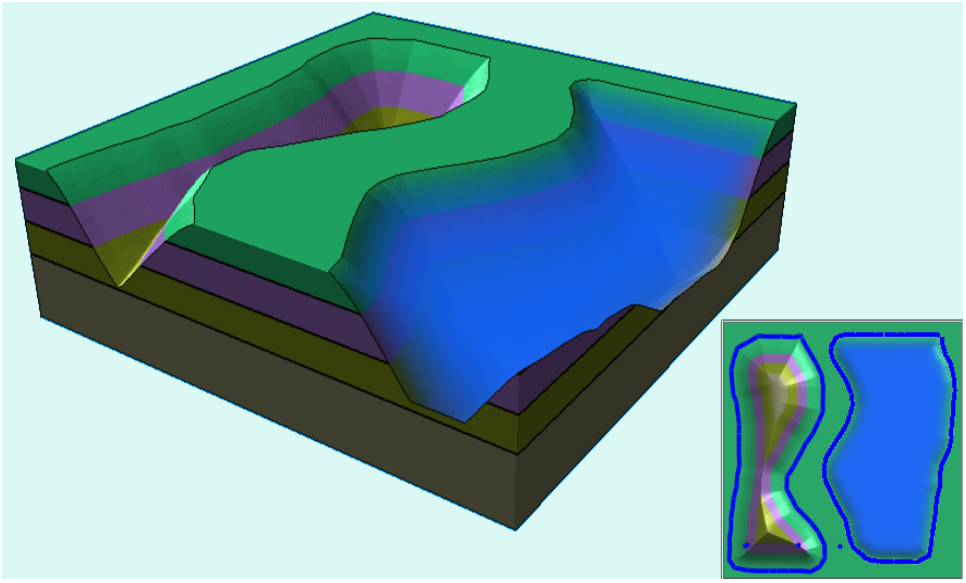

Figure 9: Two areas eroded into three layers of constant thickness. The rightmost area was filled with material, which has been assigned a translucent water appearance.

Several geological features can be generated with our method. A river with its sedimentary history can be achieved by drawing the initial and final configuration of its shape evolution, as illustrated in Figure 3. A delta representation is obtained by sketching a closed curve. Figure 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 7. A color transfer function is used to map distances to colours. Lakes and empty basins (as shown in Figure 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 10) as they are often the source of depositional material. A composition of all the previous features produces a global overview of a geological scenario, as depicted in Figure 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 2.

An important characteristic of our model representation and visualization is the ability to incorporate all the 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 2 was made to capture the different characteristics of a river system from a proximal to distal position. It has been made from interpreted river footprints found in 3D seismic datasets

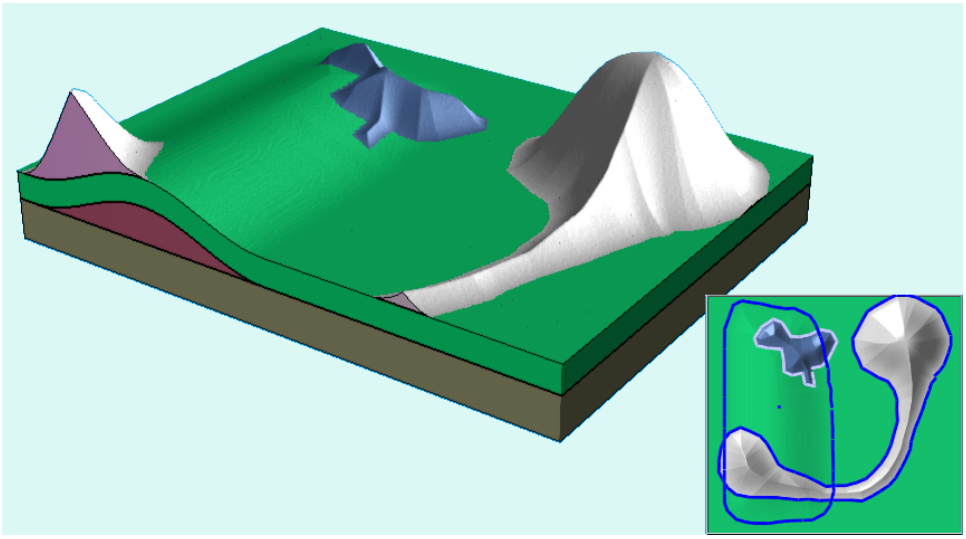

Figure 10: Layer-cake model with few elements, generated by four strokes. One is a point for specifying the green constant layer. The distance-based height calculation results in mountain-like features with a varying mountain ridge height.

of the subsurface of the Barents Sea ([KHHLG12]). The schematic block diagrams were drawn in conventional 2D vector graphics software, based on information gathered from the seismic datasets and drilled wells. The illustration highlights the cross-sectional and plane-view characteristics of this depositional environment. 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 finalising, it cannot be easily altered. On the contrary, our approach gives an automated and interactive way of drawing the diagram that speeds up the drawing process and makes subsequent alterations less laborious. Manoeuvrable planes, both cross-sectional and plane, additionally give 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). 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 sketch-based interface is simple and quick to use. For instance, the examples on the right side of Figure 2 were created in about an hour, while the hand-made image on the left, required a full day. 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 block-diagram drawings (in 2D drawing software).

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 with an *NVIDIA GeForce GTX 580* GPU.

5 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 co-author of the article.

Limitations Our method results in a $2\frac{1}{2}D$ representation, because we define our values on $2D$ regular grids. For that reason, a layer cannot fold back onto itself. Hence, there are geological scenarios that we are not able to create. Concerning rivers and deltas, we can cover most of the situations encountered in nature, since they originate from erosional and depositional processes that mostly involve the vertical direction. This is not a serious limitation when modelling depositions, as overhangs are not common. It is however possible to represent overhanging objects by representing them with several layers.

Future Work The calculation of the relative and the absolute layers is parallelizable and could in a future version be performed on the GPU. A natural extension of this work could incorporate other features, like faults, and support co-rendering of the model together with underlying seismic dataset if it exists. In addition, it would be possible to import real landscape heightmaps and combine with user defined changes to the initial geomorphology. Another improvement to our models would be given by including the technique of Zhu et al. [ZIH+11] to enrich channels with animated visualization of flow.

References

[ABPS12] Ronan Amorim, Emilio Vital Brazil, Daniel Patel, and Mario Costa Sousa. Sketch modeling of seismic horizons from uncertainty. In *Proceedings Symposium on Sketch-Based Interfaces and Modeling*, pages 1–10, 2012. 2

[BBZ09] Wang Baojun, Shi Bin, and Song Zhen. A simple approach to 3D geological modelling and visualization. *Bulletin of Engineering Geology and the Environment*, 68(4):559–565, 2009. 2

[BF01] Bedrich Benes and Rafael Forsbach. Layered data representation for visual simulation of terrain erosion. In *Proceedings Spring Conference on Computer Graphics*, pages 80–85, 2001. 2

[Bri93] J.S. Bridge. The interaction between channel geometry, water flow, sediment transport and deposition in braided rivers. In *Braided Rivers*, volume 75, pages 13–71. Geological Society Special Publication, 1993. 1

[BTHB06] Bedrich Benes, Vaclav Tesinsky, Jan Hornys, and Sanjiv K. Bhatia. Hydraulic erosion. *Computer Animation and Virtual Worlds*, 17(2):99–108, 2006. 2

[CDM+02] Barbara Cutler, Julie Dorsey, Leonard McMillan, Matthias Müller, and Robert Jagnow. A procedural approach to authoring solid models. *ACM Transactions on Graphics*, 21(3):302–311, July 2002. 2

[Cha74] G. Chaikin. An algorithm for high speed curve generation. *Computer Graphics and Image Processing*, 3:346–349, 1974. 3.3

[CVDW12] T J Coulthard and M J Van De Wiel. Modelling river history and evolution. *Philosophical Transactions of A Mathematical, Physical and Engineering Sciences*, 370(1966):2123–2142, 2012. 2

[GB07] M. R. Gani and J. P. Bhattacharya. 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*, 77:284–302, 2007. 1

[GMSe09] James Gain, P. Marais, and W. Straß er. Terrain sketching. In *Proceedings Symposium on Interactive 3D Graphics and Games*, pages 31–38, 2009. 2

[HGA+10] Houssam Hnaidi, Eric Guérin, Samir Akkouche, Adrien Peytavie, and Eric Galin. Feature based terrain generation using diffusion equation. *Computer Graphics Forum*, 29(7):2179–2186, 2010. 2

[Hin12] M. Hinderer. From gullies to mountain belts: A review of sediment budgets at various scales. *Sedimentary Geology*, 280:21–59, 2012. 1

[KHHLG12] T. Klausen, W. Helland-Hansen, I. Laursen, and R. Gawthorpe. Fluvial geometries and reservoir characteristics of a triassic coastal plain system in the norwegian barents sea. *AAPG Search and Discovery*, 2012. 4

[LJ03] Alan Lemon and Norman Jones. Building solid models from boreholes and user-defined cross-sections. *Computers & Geosciences*, 29(5):547–555, 2003. 2

[LNP+13] Endre M. Lidal, Mattia Natali, Daniel Patel, Helwig Hauser, and Ivan Viola. Geological storytelling. *Computers & Graphics*, 37:445–459, 2013. 2

[PGMG09] Adrien Peytavie, Eric Galin, Stephane Merillou, and Jerome Grosjean. Arches: a Framework for Modeling Complex Terrains. *Computer Graphics Forum (Proceedings Eurographics)*, 28(2):457–467, 2009. 2

[PS03] Szczepan J. Porębski and Ronald J. Steel. Shelf-margin deltas: their stratigraphic significance and relation to deepwater sands. *Earth-Science Reviews*, 62:283–326, 2003. 1

[Rea96] H.G. Reading. *Sedimentary Environments: Processes, Facies and Stratigraphy*. John Wiley & Sons, 704pp, 1996. 1

[SBBK08] O. Stava, B. Benes, Matthew Brisbin, and J. Krivanek. Interactive terrain modeling using hydraulic erosion. In *Proceedings ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pages 201–210, 2008. 2

[SGHS98] Jonathan Shade, Steven Gortler, Li-wei He, and Richard Szeliski. Layered depth images. In *Proceedings of SIGGRAPH '98*, pages 231–242. ACM, 1998. 3.2

[ZIH+11] Bo Zhu, Michiaki Iwata, Ryo Haraguchi, Takashi Ashihara, Nobuyuki Umetani, Takeo Igarashi, and Kazuo Nakazawa. Sketch-based dynamic illustration of fluid systems. In *Proceedings SIGGRAPH Asia*, SA '11, pages 134:1–134:8, 2011. 5