



**HAL**  
open science

## **Sculpting Mountains: Interactive Terrain Modeling Based on Subsurface Geology**

Guillaume Cordonnier, Marie-Paule Cani, Bedrich Benes, Jean Braun, Eric Galin

► **To cite this version:**

Guillaume Cordonnier, Marie-Paule Cani, Bedrich Benes, Jean Braun, Eric Galin. Sculpting Mountains: Interactive Terrain Modeling Based on Subsurface Geology. IEEE Transactions on Visualization and Computer Graphics, 2018, 24 (5), pp.1756-1769. 10.1109/TVCG.2017.2689022 . hal-01517343

**HAL Id: hal-01517343**

**<https://hal.science/hal-01517343v1>**

Submitted on 3 May 2017

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Sculpting Mountains: Interactive Terrain Modeling based on Subsurface Geology

Guillaume Cordonnier, Marie-Paule Cani, Bedrich Benes, Jean Braun, Eric Galin

**Abstract**—Most mountain ranges are formed by the compression and folding of colliding tectonic plates. Subduction of one plate causes large-scale asymmetry while their layered composition (or stratigraphy) explains the multi-scale folded strata observed on real terrains. We introduce a novel interactive modeling technique to generate visually plausible, large scale terrains that capture these phenomena. Our method draws on both geological knowledge for consistency and on sculpting systems for user interaction. The user is provided hands-on control on the shape and motion of tectonic plates, represented using a new geologically-inspired model for the Earth crust. The model captures their volume preserving and complex folding behaviors under collision, causing mountains to grow. It generates a volumetric uplift map representing the growth rate of subsurface layers. Erosion and uplift movement are jointly simulated to generate the terrain. The stratigraphy allows us to render folded strata on eroded cliffs. We validated the usability of our sculpting interface through a user study, and compare the visual consistency of the earth crust model with geological simulation results and real terrains.

**Index Terms**—Terrains, Mountains, Interactive Design, Geology.

## 1 INTRODUCTION

VIRTUAL terrains are essential to many applications, from real-time flight simulators and games to movies. They are also used in geology by scientists who study interactions between surface processes (such as erosion) and tectonics, by educators, and by scientists targeting public outreach.

While geologists have been gradually advancing the simulation of mountain formation processes, models are either focused on the mechanics of crustal deformation [1] or on computing the Earth’s surface response to tectonic uplift [2]; their purpose is not to generate visually compelling 3D terrains at interactive rates. In contrast, terrain generation methods in Computer Graphics already achieve visually realistic results and recent advances allow for interactivity and user control. However, Computer Graphics methods often fail to capture large-scale landforms commonly observed in nature. One reason for that is that most Computer Graphics methods focus on modeling the surface of terrains, and do not attempt to reproduce the geometry of folds, faults, or the uplift systems that shape mountainous landscapes and which, eventually, produce patterns that are observed at various scales, from eroded cliffs to regularly spaced valleys. Moreover, the lack of stratigraphic information (*i.e.*, information on how the characteristics of rock change with depth) in scanned real terrain databases prevents example-based terrain synthesis methods from capturing many of

these important features.

We present the first volumetric model for the interactive generation of visually-compelling, large-scale terrains aimed at Computer Graphics applications. We address the long-standing challenge of providing the user with interactive control of large-scale terrains, such as shaping mountain ranges, while maintaining visual consistency with phenomena observed in nature, including the modeling of multi-scale folds. We based our work on a well-known fact in the geological community that mountain ranges are formed by several interdependent phenomena triggered by tectonic forces, which, as we demonstrate here, can be efficiently modeled by using knowledge-based, phenomenological simulations. We coin the term *geologically-inspired simulation* for simulations that exploit first-order phenomenological knowledge from geology.

When tectonic plates<sup>1</sup> collide, the Earth’s crust is thickened in an incompressible manner, which causes the surface to be uplifted by isostasy: considering that tectonic plates float on a much less viscous and more dense part of the upper mantle like an iceberg floating in the ocean, any variation in crustal thickness must be accompanied by relative vertical motion of the plate’s surface. Collision often leads to the formation of faults or dipping shear zones that cut through the entire crust. Within the uplifting region, shortening is accommodated by smaller scale folding. The wavelength of the folds depends on the stratigraphy of the crust, *i.e.*, the thickness of layers previously deposited at the surface of the Earth and their relative competency. The surface uplift leads to *fluvial erosion* resulting in the formation of dendritic drainage systems and valleys. The rate of river incision is controlled in part by the competency

- G. Cordonnier and M.-P. Cani are with Univ. Grenoble Alpes, CNRS (LJK), and Inria, Grenoble, France.
- B. Benes is with the Department of Computer Graphics, Purdue University, West Lafayette, IN, 47907-2021.
- Jean Braun is with the GFZ German Research Center for the Geosciences and University of Potsdam, Germany
- E. Galin is with the LIRIS Lab, Université Lyon 2 and CNRS, Lyon, France

1. See appendix A for definitions of geological terms.

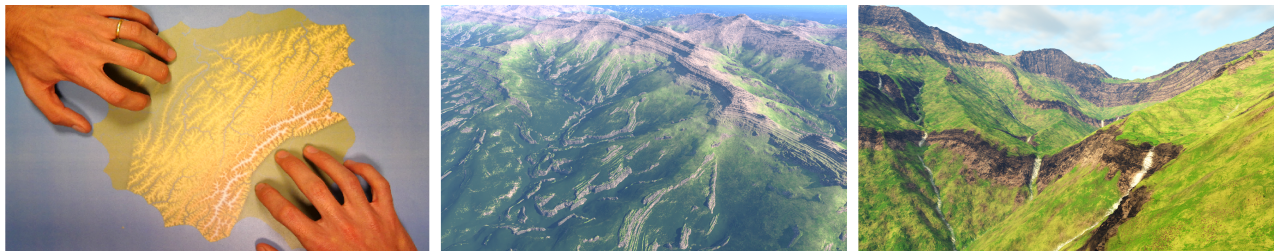


Fig. 1. The user controls the motion of tectonic plates by hand gestures (left). We automatically generate a volumetric terrain model that is folded and eroded (large scale view in the middle). The small scale view of the eroded cliffs (right) shows the folded strata that appears on cliffs.

of rocks, which means that the stratigraphy of the crust strongly influences the shape of river profiles and valley walls and may lead, for example, to the formation of local cliffs or ledges that stand out in the landscape.

Our main contribution is a novel, *geologically-inspired* deformable model that mimics the Earth's crust under tectonic forces. While we do not claim to reproduce complexity of geological models that incorporate formal mechanical calculations coupled with sophisticated erosion models that require heavy computations [3], we provide an interactive framework for intuitively controlling the creation of plausible terrains in Computer Graphics. Our framework could be also used for interactive illustration purposes in geology. The geological approximations are justified by the interactivity of our sculpting system and by the targeted result: a geologically-plausible large scale eroded terrain surface that hides most of the higher order tectonic processes.

Our multi-layered (or stratigraphic) model [4] of crustal material captures the combined effect of compression and folding. We consider that, at the scale of the crust, rocks behave in a ductile (or viscous) manner and deform by uniform shortening. This is similar to the thin-sheet approximation used in geodynamical models [5], based on the observation that -deep in the Earth's crust- rocks deform by ductile creep at very low strain rate. In the upper part of the crust faulting and folding take place. We use a procedural model to add multi-scale folds triggered by the compression field, based on the specific features of the chosen subsurface stratigraphy. The resulting deformation model is used to compute the uplift of the surface. This uplift is then used in a real-time erosion model, enabling interactive visualization of the resulting sculpting of the surface by erosional processes. The additional stratigraphic details of our model (smaller scale layers) enable us to generate multi-scale folds and produce the final, high quality terrain models used for final rendering.

Our method allows hands-on interactive design of mountain ranges: users can quickly draft and then progressively refine large scale terrains through gestural control, while ensuring that the generated mountain ranges are consistent with first-order geological processes. Figure 1 shows an example of an interactive session where the user sculpts and refines multi-layered earth-crust material as if it were plasticine, using displacement gestures applied using finger interaction on a multi-touch display. The first gestures (Figure 1, left) define the tectonic plates and their relative motion. The resulting collision causes surface uplift and the growth of mountains, where the speed of growth

is stored in an uplift map. The combination of the uplift and of an existing fluvial erosion simulation is visualized at interactive rates, enabling the user to refine his/her design. High quality rendering of the resulting terrain at different scales (Figure 1, center and right) shows the effect of a non-uniform input stratigraphy on the shape of valley sides.

We validated our method by comparing our results to simulations from geology and to real terrain data. We show that various phenomena appearing at different scales are correctly captured such as the asymmetric large-scale shape of mountain ranges or the small scale, stratigraphically-controlled folds that appear along eroded cliffs. We also conducted a user study with artists and non-artists which confirmed the usability of our sculpting tool.

## 2 RELATED WORK

In this section, we discuss algorithms for terrain modeling relevant to our method and then review the use of sculpting metaphors for interactive shape design. For a more comprehensive review of terrain and virtual world modeling techniques, we refer the reader to [6] and [7].

**Procedural generation** techniques, such as fractals and noise-based generators [8], are efficient multi-resolution models for generating terrains. They hold a prominent role in many applications. Despite a few attempts to force fractal generation to fit user-defined constraints [9], [10], the common problem of procedural terrain generators is their lack of direct control. In contrast other procedural methods model terrains from a set of 3D feature curves used to specify ridges or river networks [11], [12], with, however, little help from the system to improve realism. G enevaux *et al.* [13] introduce a method that builds large terrains by combining river sections and terrains into a single geologically-based model. Another function-based approach [14] defines terrains as hierarchical construction trees, combining primitives with blending, carving, and warping operators. Gu erin *et al.* [15] introduce a procedural method to represent terrains as elevation functions defined as a sparse combination of primitives built from linear combinations of landform features stored in a dictionary. These features can be extracted either from real world datasets or procedural primitives, or from any combination of multiple terrain models. A common problem with these methods is their lack of control. If the results do not match the design goals, the user cannot easily refine his/her design and needs to re-launch the generation of the procedural terrain with different parameters.

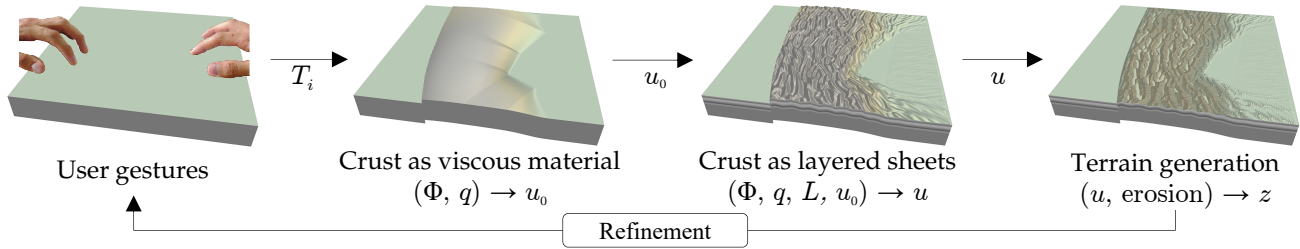


Fig. 2. User gestures define tectonic plate movement that is used to simulate earth crust as if it was made of a viscous material. The compression and thickness change first yields a coarse uplift map  $u_0$  and then a thin layered strata that produces a refined uplift map  $u$ . Fluvial erosion then produces the final terrain elevation  $h$ . The overall simulation is interactive and allows for real-time authoring by hand gestures.

**Example-based methods** explore the reuse of existing terrains in order to improve realism. Many example-based methods are inspired by texture synthesis; for example the method of [16] exploits a user-painted coarse image to control the combination of patches extracted from real terrain data. Terrain samples can also be combined along user-drawn mountain ranges [17], or can be edited by re-sketching the silhouettes of mountains from a first person viewpoint [18]. Although these methods improve user control while maintaining small scale features from the original terrain data, they generally do not capture large scale land-form features commonly observed in real landscapes and do not claim geological realism.

**Erosion simulation** approaches can be applied to any terrain model to improve the realism of land-form features. Thermal erosion was introduced by Musgrave *et al.* [19], later extended by approaches tackling hydraulic erosion [20], [21], accelerated by using computations on the GPU [22], [23], applied to layered terrains [4], or controlled in an augmented reality environment [24]. These methods accurately capture small scale terrain features but require a terrain model to be eroded, which can induce large scale inconsistencies. In contrast, the work of [25] proposes a geologically-inspired erosion model that achieves the generation of large scale, plausible terrains by combining fluvial erosion with mountain growth. The user controls the process through the input of an uplift map representing the growth rate of mountains. However, the user is not given any help in designing a plausible uplift map, reflecting the usual features of mountain ranges. In our work, we build on this erosion method and combine it with an intuitive, gesture-based method for designing the uplift map, namely sculpting earth-crust material and allowing mountain ranges to grow through the action of plate tectonics.

**Plate tectonics simulations** has been performed and studied for a long time in the geological community and many sophisticated thermo-mechanical models were proposed [1], [3]. These models require, however, intensive computations. We are not aware of any existing model enabling an interactive 3D simulation of these phenomena. To the best of our knowledge, interactive 3D modeling of plate tectonics was only used in [26] to generate a procedural terrain from a 2D vector map of mountain summits. There, the terrain is divided into tectonic plates by using Voronoi-based tessellation and an image processing method is used to generate simple multi-scale folds along plate borders, without taking any layered earth-crust model into account. In contrast,

we introduce the first 3D interactive simulation of plate tectonics, thanks to a deformable approach that couples a viscous model with multi-scale folding of multi-layered rocks. Our model provides a more accurate simulation of plate collision and folding processes.

**Sculpting** has long been one of the main expressive modeling metaphors [27]. In addition to physically-based virtual plasticine [28], sculpting systems have proven to be successful for editing as rigid as possible objects [29] or shapes with nested features [30], and structured shapes, such as architectural models [31]. Peytavie *et al.* [32] introduced a hybrid model combining material piles and an implicit representation to interactively sculpt 3D terrain models featuring overhangs and caves. Their approach does not rely on geology, however. Other approaches also use tactile devices [33] or sketches [34] to illustrate sub-surface geology.

We present the first application of sculpting to geologically-based earth crust material, enabling users to progressively design mountain ranges through gesture-based interaction on a multi-touch table. As in most sculpting systems, deformation automatically propagates in a plausible way in-between the points the user manipulates; here, through the setting of tectonic, earth-crust motion. An important difference from existing virtual-clay or *plasticine* material is that the earth crust plates are thin shells. While they deform as viscous material and keep a constant volume under compression, they also fold in compressed regions due to their multi-layered nature.

### 3 OVERVIEW

In this section we describe the geological background used as the basis for our approach and the choice of our algorithms and data structures.

#### 3.1 Plate tectonics in geology

Continental collision is often the end product of the closure of an oceanic basin driven by subduction of the oceanic plate beneath one of the two continents. When continental plates collide, subduction of the lower part of one plate continues (left plate in Figure 3), while the upper layer (the continental crust) made of lighter, granitic rock resists subduction and experiences compression. This compression is accommodated by the formation of a main crustal-scale fault or shear zone, located above the stable, non-subducting plate, that accumulates the deformation by thrusting [1]. In other situations, for example where the crust is hotter,

the deformation is more diffuse and the crust deforms as a thin viscous layer [5]. In all cases, this causes uniform uplift of the crustal block and we model this growth by approximating the crust as a viscous material (Section 4).

Simultaneously, this block deforms internally by smaller scale faulting and folding. Due to past tectonic events and/or the deposition of sedimentary rocks at the Earth's surface, the continental crust is commonly composed of a number of horizontal rock layers of different mechanical properties. On top of the basement, made of metamorphic and igneous rock, layers of sedimentary rock are deposited over millions of years, due to the erosion of older mountains or to the deposition of carbonate shells in ancient oceans. When compressed during plate collisions, the difference in mechanical properties between the sedimentary layers makes them fold rather than thicken. We compute the multi-scale folding from a thin sheet approximation of the crust (Section 5).

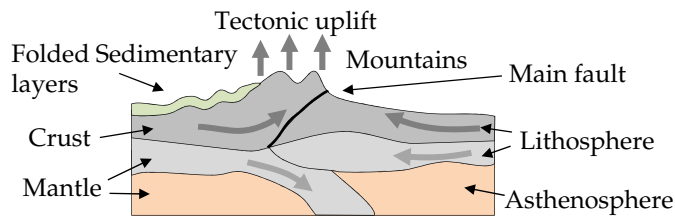


Fig. 3. Colliding tectonic plates: plate experiencing subduction (left) with uplift and folding of crust material while the lithosphere plunges down, main fault (middle), and stable plate (right).

As a result, mountain ranges are typically asymmetrical, with uplift and folding leading to high topography and steep surface slopes on one side of the mountain where the main structure is located, and more subdued elevations and lower slopes on the other side (*i.e.*, the right of Figure 3). The fact that continental crust is made of a number of different rock layers causes different fold scales to appear (top left of Figure 3). The surface slopes resulting from these complex patterns of surface uplift lead to concomitant surface erosion controlled by the rate of fluvial incision [2] which, in turn, results from the shear stress imposed by debris transported by fast flowing rivers on the bedrock (Section 6).

### 3.2 Geologically-inspired interactive simulation

Our goal is to enable the sculpting of large scale terrains (hundreds of kilometers) at interactive rates, where the user's time corresponds to millions of years in real temporal domain. The corresponding simulated time is around 5 million years in our examples.

**User input gestures** on a multi-touch input device define the shape of tectonic plates and put them in motion (see Figure 2 and the accompanying video). The user first places a reference hand on the table to define the first tectonic plate, serving as an anchor and intended to be stationary. He/she then moves the second hand in contact with the table to define and set the motion of the second plate (defined as a Voronoi region associated with the second hand) and can repeat this gesture to add new plates. Each time a plate is created, crust material is considered to move rigidly in the convex hull of finger positions, and finger motion is

interpolated to define a single, rigid transformation, interpreted as the rate at which the plate moves per time unit (Section 4). Assuming that, at this scale, *i.e.*, when geological temporal scales are remapped into a few seconds, the Earth's crust deforms like a soft, viscous material and inertial forces can be neglected, this motion is unaffected by the collision process and can therefore be assumed to remain constant over time – except when the user applies further edits. The output is a transport field denoted as  $\Phi$ .



Fig. 4. The initial stratigraphy: rock layers  $L_i$  have different mechanical properties, whereas sub-layers differ by erosion parameters only.

**Subsurface stratigraphy** (Figure 4) is defined prior to the simulation using a multi-layered, volumetric model of the Earth's crust:  $n$  layers of rock  $L = \{L_i\}$ ,  $i = 1, \dots, n$  are defined with corresponding thicknesses  $0.1 \leq \tau_i \leq 5$  km and viscosity  $\mu_i$  (a measure of the competence of the rock). They represent variations in rock type and mechanical strength at a variety of scales. Thinner layers are used to represent sedimentary material located above thicker, basal rock layers that represent the layering of the crust, for example separating it into a quartz-rich upper crust and a feldspar-rich lower crust. In our implementation, layer thicknesses are spatially uniform, although extending our method to have them vary locally would be straightforward.

To increase visual realism, each layer can be subdivided into 1 – 10 sub-layers of varying resistance against erosion. This represents the fractal nature of natural stratigraphy where large competence contrast is likely across thick layers that are themselves composed of thinner layers of reduced competence contrast.

**Geologically-inspired simulation** consists of a simulation loop that uses three consecutive actions to compute the terrain deformation at each time step (Figure 2).

- 1) From the user-defined transport field  $\Phi$ , we compute the elevation rate (or *uplift*)  $u_0$  of the terrain caused by the compression between tectonic plates, assuming that the earth crust behaves as an incompressible viscous material in the considered geological time step (Section 4).
- 2) We then compute the uplift field  $u_i$  of each layer of rock according to the associated multi-scale folding processes. This is performed using an efficient procedural method based on results from the theory of folding of layered materials (Section 5).
- 3) We apply fluvial erosion and a landsliding algorithm [25] to shape the terrain during mountain growth due to uplift (Section 6). This results in a time dependent surface elevation  $z(\mathbf{p}, t)$ , in which we keep track of the original rock layer at each node position, for rendering purposes.

**Time steps:** Recall that we are sculpting *animated material*: the mountains grow and are eroded while they are sculpted. Therefore, there are two temporal loops: the *geological simulation* loop based on the current user input (which gives us a stationary compression field and thus stationary uplift and folding, then combined with erosion), and the *user interaction* loop (Figure 2). The compression field, the folds, and then the uplift need to be recomputed only when a new interaction takes place (*e.g.*, when the motion of the plates is modified). Even when not recomputed, they are used to update the terrain elevation at each frame. The geological time simulation step (Section 4) is not the same as the time-step for any of these loops: it is a theoretical time step used for computing the stationary compression field from the current plates motions. Its value is not important since it expresses a stationary situation.

#### 4 EARTH CRUST AS A VISCOUS MATERIAL

Shortening of the crust is the first phenomenon to be accounted for when tectonic plates collide. The crust at this scale can be assumed to behave as a viscous material, resulting in thickness changes and uplift. To capture this phenomenon, we use a 2D physical simulation that computes a *transport field*  $\Phi(\mathbf{p})$  in the region of influence between plates. The transport field is a map  $\Phi : R^2 \rightarrow R^2$  that describes where each point moves subject to the tectonic forces. It is coupled with a geometric volume preservation method to generate the uplift.

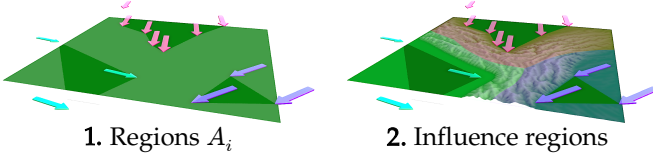


Fig. 5. Gestural control: Three moving plates are created and refined by using successive hand motions and their rigid part are set to the convex hull of the interacting fingers (1). The region of influence of each plate is defined as the part of the plane that is closest to its rigid part (2).

##### 4.1 Moving plates creation

Every plate is characterized by its supporting region  $A_i$  where the earth crust displacement is rigid. The associated transformation  $T_i$  (a rotation and a translation) represents the rigid displacement of the underlying crust material during a simulation step. While  $A_i$  does not move over time and  $T_i$  is constant, the continuous displacement of crust material causes surface uplift and mountain growth in colliding regions.

To define a new plate, the user moves his/her hand in contact with the touch table. We provide visual feedback by displaying an arrow from the initial position  $\mathbf{c}_k^i$  of each finger to its final position, *i.e.*, when contact with the table ends. The support region  $A_i$  is defined as the convex hull of the initial finger positions  $\mathbf{c}_k^i$ . Without loss of generality, the vector  $\Phi(\mathbf{c}_k^i) - \mathbf{c}_k^i$ , corresponding to each arrow, is interpreted as the motion of the underlying earth crust during a given geological simulation step  $\Delta t$ .

Since the user gesture may not be fully rigid, the plate's rigid transformation  $T_i$  is computed from the user defined

set of transport directions at control points  $\Phi(\mathbf{c}_k^i)$  by using least squares minimization:

$$\operatorname{argmin} \sum_k \|\Phi(\mathbf{c}_k^i) - T_i \mathbf{c}_k^i\|^2.$$

The term  $\Phi(\mathbf{c}_k^i)$ , and the corresponding arrows are then updated from  $T_i$  so that the arrows always depict the rigid motion of the crust material under the rigid part of the plate during a geological step  $\Delta t$ . The user can use extra gestures to add new arrows and edit  $T_i$  accordingly.

When crust material moves away from the regions  $A_i$  to collision regions, the assumption of rigid motion is no longer true. We call the *influence region* of each plate the portion of the crust closest to the associated  $A_i$ . We interpolate the displacements  $T_i$  to compute the transport field  $\Phi$  in those regions of influence.

##### 4.2 Viscous compression

The transport field  $\Phi$  describes the local movement of earth crust material during  $\Delta t$  under the action of tectonic forces, and is computed as the interpolation of the transformations  $T_i$  given by the user. At the space and time scale of our simulation, the earth crust can be approximated as a thin plate of viscous material that shortens under the action of plate motion and collision (Figure 5 (b)) [5]. At this scale rock deformation is isochoric and the viscous deformation can be assumed to be incompressible such that horizontal shortening yields thickness changes.

**The transport field.** To simulate the 2D compression, we adapt an existing model of viscous virtual plasticine [28]. The amount of viscous crust material  $q$  is stored in a 2D regular grid, to be transported from cell to cell through  $\Phi$ . The grid is defined by a step  $dx$  and an orthogonal frame  $(\mathbf{e}_x, \mathbf{e}_y)$ . Each grid cell is represented by its center  $\mathbf{x}_i$ , and  $q_i$  represents the volume of crust rock inside the cell. The area of the grid cell is denoted by  $a_{cell}$ .

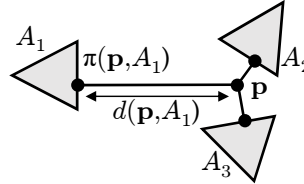


Fig. 6. Transport at  $\mathbf{p}$  is interpolated between support regions  $A_1$ ,  $A_2$  and  $A_3$ , depending on the projection  $\pi(\mathbf{p}, A_i)$  of  $\mathbf{p}$  onto  $T_i$  and distance  $d(\mathbf{p}, A_i)$  between  $\mathbf{p}$  and  $T_i$ .

Inside the transformed rigid area  $T_i(A_i)$  the inverse of the transport is set to the rigid transformation

$$\Phi^{-1}(\mathbf{p}) = T_i^{-1} \mathbf{p}. \quad (1)$$

Outside the region  $A_i$ , the field  $\Phi^{-1}$  is computed by using an interpolation based on a distance weighting operator. Let  $d(\mathbf{p}, A_i)$  denote the distance between  $\mathbf{p}$  and region  $A_i$  and  $\pi(\mathbf{p}, A_i)$  the projection of  $\mathbf{p}$  onto the region  $A_i$  (Figure 6). We define:

$$\Phi^{-1}(\mathbf{p}) = \mathbf{p} + \sum_i \frac{T_i^{-1} \pi(\mathbf{p}, A_i) - \pi(\mathbf{p}, A_i)}{d(\mathbf{p}, A_i)^e} / \sum_i \frac{1}{d(\mathbf{p}, A_i)^e} \quad (2)$$

and we set the exponent  $e = 0.9$  to get a smooth falloff of the influence of the distance.

**Crust material transportation** We use  $\Phi^{-1}$  to compute the amount of crust material  $q$  in grid cells after the geological step  $\Delta t$ , *i.e.*, when plate material is transported from the tip to the end of the arrows (Figure 5).

Let  $q_0$  be the initial amount of material in each grid cell. In order to obtain the amount of material  $q(\mathbf{x}_i)$  in a cell  $c_i$  at the end of the plate’s motion, we use the inverse transformation  $\Phi^{-1}$  to compute the original positions of the four corners of  $c_i$  before motion, which defines a quadrangle of area  $a_i$ . Then we compute the value of  $q$  in the cell  $c_i$  after the simulation step:  $q(\mathbf{x}_i) = q_0 a_i / a_{cell}$ . The initial amount of material is factored later to compute the uplift, so its value does not have any physical meaning. We simply set  $q_0 = 1$ .

### 4.3 Uplift from thickness Changes

Crustal deformation is isochoric, *i.e.*, crust material retains a constant volume over time. The local thickness of the crust is proportional to the volume of crust rock inside the associated cell and to the amount of material  $q$  in the 2D grid. Therefore, variations in the amount of material provide a map of thickness change rates:  $dq/dt = (q - q_0)/\Delta t$ .

Since our simulation method does not capture the main fault formation process and the resulting discontinuity in altitude (Section 3.1), we model them by a procedural approach. We define the fault as an offset of the boundary of the compressed region on the stable plate side that is computed as an iso-level of  $q$ . We then define the uplift due to compression as  $u_0 = 0$  on the stable side of the fault and  $u_0 = \alpha dq/dt$  elsewhere. The coefficient  $\alpha$  is a scaling coefficient used to set the uplift to the appropriate geological time scale; in our system we set  $\alpha$  so that the uplift is  $5.0 \cdot 10^{-4} \text{m/y}$  on average. This approach captures the asymmetric nature of the earth crust compression. The resulting uplift map  $u_0$  is further improved by incorporating folds, as described next.

Note that this procedure neglects two potentially important aspects of crustal shortening. First, we neglect isostasy by imposing that the compression-induced shortening results in surface uplift only. In most mountain belts, shortening also results in a vertical depression of the base of the crust (and of the plate). Second, we do not prescribe a dip to the main fault. In nature all compressive faults (or thrusts) dip at an angle of a few tens of degrees with respect to the surface. The consequence of our approximation is that we do not accurately compute the additional horizontal velocity field that arises from the deformation process. We note, however, that such an approximation is also commonly used in the geological community as the effects of horizontal advection of landforms have only been recently investigated [35].

## 5 EARTH CRUST AS LAYERED SHEETS

The deformation and shape of mountain ranges are also controlled by folding of layers within the crust under compression. More precisely, the crust is composed of several layers of rocks of varying thickness and strength. Complex multi-scale folding occurs, where small folds appear on top

of larger ones. Since simulating this complex folding process would be computationally demanding, we use a procedural model to capture this well understood behavior and then rectify the uplift map  $u_0$ , which only captures the viscous behavior of the crust.

### 5.1 Folding of layered materials

Folding is a mechanical instability that results from the growth of stress imbalance at the interface between layers with different competence. Researchers in both geology [36] and Computer Graphics [37] have studied this phenomenon. They rely on the observation that the wavelength of folds  $\lambda$  of a layer  $k$  of thickness  $\tau_k$  and viscosity  $\mu$  embedded in a viscous material are correlated, which can be expressed by the following equation [38]:

$$\lambda_k \sim (\tau_k \tau_{k-1})^{1/2} (\mu_k / \mu_{k-1})^{1/6}. \quad (3)$$

Our work relies on this theoretical law as it provides a good approximation of the fold formation process. Similar expressions would be obtained assuming an elastic or viscoelastic rheology for the layers; the important point is that we make the wavelength of the folding instability proportional to the thickness of the layer(s) and their competence contrast (viscosity or elastic properties), both of which can be controlled as input parameters.

The user can define the wavelength of the folds that directly controls the folding effects. The fold thickness is computed automatically.

### 5.2 Procedural fold generation

Although folds of appropriate wavelengths could be set to emerge from layered sheet simulation [37] or could be generated by combining elastic thin sheet material [39] with smaller scale folds [40], we generate the folds procedurally on top of the viscous simulation (Section 4). This allows us to achieve interactive rates, which is key for seamless content creation with sculpting gestures. Moreover, directly reusing wavelengths from Equation (3) is sufficient to ensure visually plausible folds. Therefore, we introduce a procedural modeling method for multi-scale folds.

Our method forces the folds to follow a set of lines orthogonal to the compression direction. We constrain those lines to be separated by an average distance  $\lambda$  that is the theoretical wavelength. The lines are chosen based on the compression direction extracted from the transport field  $\Phi$ . This choice corresponds to the places of maximum folding (called *top skeleton* below), between which the folds are interpolated. This provides visually plausible results consistent with the expected fold wavelength and height.

**Extracting the compression field.** We extract the folding direction and amplitude from the crust *compression field*  $C$  by first converting the inverse transport  $\Phi^{-1}$  given by Equations (1) and (2) into a deformation tensor field  $\mathbf{F}$ :

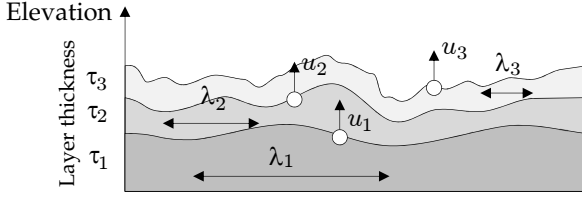
$$\mathbf{F} = (\mathbf{I} - \nabla(\Phi^{-1} - \mathbf{I}))^{-1}.$$

The stretch tensor field  $\mathbf{U}$  is computed by taking the square root of the Cauchy-Green deformation tensor [40]. In our

case, this can be written as  $\mathbf{U} = \sqrt{\mathbf{F}^T \mathbf{F}}$ . For every cell, the  $2 \times 2$  matrix  $\mathbf{U}_i$  can be diagonalized:

$$\mathbf{U}_i = \nu_1 \mathbf{e}_1 \mathbf{e}_1^T + \nu_2 \mathbf{e}_2 \mathbf{e}_2^T,$$

Vectors  $\mathbf{e}_1$  and  $\mathbf{e}_2$  are the unit eigenvectors in the plane and  $\nu_1 \leq \nu_2$  with real positive eigenvalues. We compute a *compression field*  $C$  for the crust, defined in every cell  $i$  by:  $C_i = (1 - \nu_1) \mathbf{e}_1$  if  $\nu_1 < 1$  and  $C_i = 0$  otherwise. The vector  $\mathbf{e}_1$  is the direction of compression and  $1 - \nu_1$  represents its strength. Geological events that would cause some local extension of crust are beyond the scope of this paper and are not considered.



Folds within the different layers of the continental crust

Fig. 7. The different fold layers are stacked and their uplift contributions are summed to obtain the final uplift.

**Iterative fold generation.** We model the folds resulting from the layered structure by processing layers one after another (Figure 7) and summing the resulting elevation rates  $u_i$  to get the final uplift rate:

$$u = u_0 + \sum_{i=1}^n u_i. \quad (4)$$

This captures folding effects at different scales, since each  $u_i$  represents folding at the specific wavelength  $\lambda_i$  (Equation (3)) that results from the interaction between the pair of layers  $L_{i-1}$  and  $L_i$ . Folds are generated using sinusoidal interpolation [41] between skeletal curves computed among the edges of a random 2D grid, and then used to compute  $u_i$ , as described below.

**Fold skeletons.** We compute crest and bottom lines for the folds called the *top* and the *bottom* skeletons. Similar to previous fold generation methods [40], these skeletons are set to be orthogonal, in average, to the compression direction. We also make sure that they cover the whole compression region. This is done as follows:

We first generate a planar mesh  $\mathcal{M}$  independent from the terrain grid, and chosen such that the average edge length is equal to the desired wavelength  $\lambda_i$  from Equation (3).  $\mathcal{M}$  is defined as a Delaunay triangulation of a Poisson disc sampling of the terrain, with  $\lambda_i$  as average edge length.

Then, some edges of  $\mathcal{M}$  need to be selected as part of the *top skeleton*  $S_t$  (crest lines). We use the compression field  $C$  to compute  $S_t$  as follows. Let  $[\mathbf{a}, \mathbf{b}]$  be an edge of  $\mathcal{M}$  with center  $\mathbf{m}$ . For every edge surrounded by two faces  $F_0$  and  $F_1$ , we define a dual edge as an edge linking the center of an edge of  $F_0$  to the middle of an edge of  $F_1$ , both being distinct from  $[\mathbf{a}, \mathbf{b}]$ .

We define the *compression energy*  $e$  along an edge  $[\mathbf{a}, \mathbf{b}]$  to measure its alignment with the direction of compression:

$$e([\mathbf{a}, \mathbf{b}]) = \left| \frac{(\mathbf{b} - \mathbf{a}) \cdot C_i(\mathbf{m})}{\|\mathbf{b} - \mathbf{a}\| \cdot \|C_i(\mathbf{m})\|} \right|.$$

An edge with low compression energy is more likely to be in the direction of the folds. Therefore, an edge  $E$  is added to  $S_t$  if and only if its folding energy is below one of its four dual edges. It can occur that the three edges of a face of  $\mathcal{M}$  belong to  $S_t$  which would cause a singularity. We detect these cases and remove the edge of highest energy from  $S_t$ .

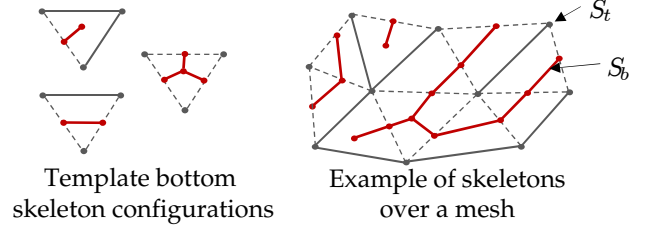


Fig. 8. Folds computation starts by choosing edges for the top skeleton  $S_t$  (gray) over a mesh  $\mathcal{M}$  (dotted gray) in the best fold direction. The bottom skeleton  $S_b$  (red) is deduced from  $S_t$  following the template (left)

The third step is to build the skeleton  $S_b$  representing the bottom parts of the folds. Figure 8 shows the template configurations that are used to generate  $S_b$  from  $S_t$ . For every triangle  $F \in \mathcal{M}$  we count the number of edges  $n_F$  of  $F$  that belong to  $S_t$  and apply the following classification. If  $n_F = 2$  we add an edge to  $S_b$ , linking the center of the edge of  $F$  that is not in  $S_t$  to the center of  $F$ . If  $n_F = 1$  we add an edge to  $S_b$  that links the center of the two edges of  $F$  that are not in  $S_t$ . Otherwise we add three edges connecting the center of one edge of  $F$  to the center of  $F$ . Note that this construction ensures that the edges of  $S_b$  remain approximately parallel to the main folds direction, i.e. to the average direction of neighboring  $S_t$  edges.

**Fold heights.** In the last step, we build the fold surface between the bottom and the top skeletons. Although the average wavelength is near the theoretical value from Equation (3), it may vary among the folds and we need to know the precise wavelength at any point in order to compute its height. Let  $d(\mathbf{p}, S_t)$  and  $d(\mathbf{p}, S_b)$  denote the distance between point  $\mathbf{p}$  and each skeleton. We define the local wavelength as:

$$\lambda(\mathbf{p}) = 2 (d(\mathbf{p}, S_b) + d(\mathbf{p}, S_t)),$$

and the local height of the fold at time  $\Delta t$  (see Figure 9) as:

$$f_i(\mathbf{p}) = \frac{\sigma_i \Delta t \cdot \|C_i(\mathbf{p})\|}{2} \left( 1 + \cos \left( \frac{2\pi d(\mathbf{p}, S_t)}{\lambda_i(\mathbf{p})} \right) \right),$$

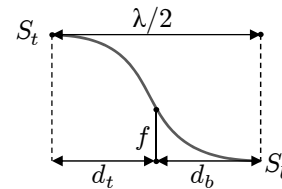


Fig. 9. The local wavelength value is calculated from the distance from top to bottom skeletons, and used to set fold height.

where  $\sigma_i$  is the folding speed. We choose it as a fraction (between 0.01 and 0.5) of the uplift  $u_0$ . Note that we add one to the usual cosine curve used to shape the fold [41] because we model the folding behavior of a rock layer that slides over some thicker layer below it, so

$f_i(x)$  should be always kept positive.



**Local fold refinement.** This step aims at complementing our procedural fold model with a local, direct editing tool that allows for authoring the ridges and valleys in a mountain range. Local edits are useful controls for designers in Computer Graphics applications such as movies and games. The user can change the direction of the folds by manually selecting edges and can also change the height of the mountain along a fold line. The advantage of this local refinement tool is that it offers direct control over the local ridges and valleys while ensuring to some extent the geological consistency of the results, since the computed fold skeletons serve as a geological guide for the user to maintain the coherency of the generated mountain.

### 5.3 Uplift update from folds

The uplift  $u_i$  resulting from the folding of the current fold pair of crust layers is defined as the folding speed  $\Delta f$ :

$$u_i = \Delta f = \frac{f(t_0 + \Delta t) - f(t_0)}{\Delta t}.$$

This value is used in Equation (4) to compute the total uplift field  $u$ , and yields terrains that fold at multiple scales.

In practice, using several different folding scales is not necessary for the interactive visual feedback during a sculpting session, since only large folds are noticeable. The other fold layers are considered only during the offline rendering of the final terrain.

## 6 TERRAIN SURFACE GENERATION

The elevation of the terrain is generated from the combined uplift and stratigraphy of the crust by applying fluvial erosion and landsliding processes. In our system, we did not include a representation of glacial erosion; to our knowledge there is currently no numerical method that can interactively simulate this process which involves the computationally demanding simulation of ice thickness and flow [42].

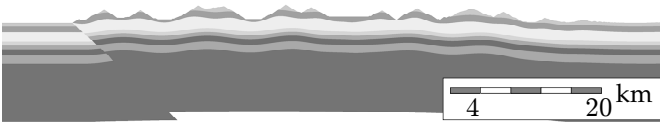


Fig. 10. Schematic rendering of a geological cross-section of a synthesized mountain range.

### 6.1 Interactive terrain generation

The terrain surface generation algorithm is based on [25] and it combines the uplift (Equation 4) with a fluvial erosion law. We use the stream power law commonly used in geomorphology that states that the rate of fluvial incision is proportional to local discharge (the flux of water in the river) and bedrock slope [43]. Discharge is assumed to be proportional to drainage area. Our method for solving this law relies on a triangle irregular network (TIN) that represents the topology of the terrain. The drainage area  $\mathcal{A}(\mathbf{p})$  and slope  $s(\mathbf{p})$  are derived for each point  $\mathbf{p}$  in the triangulation

in order to compute a rate of elevation change following the stream power law:

$$\frac{\partial z(\mathbf{p}, t)}{\partial t} = u(\mathbf{p}) - k \mathcal{A}(\mathbf{p})^m s(\mathbf{p})^n, \quad (5)$$

where coefficients  $k$ ,  $m$ , and  $n$  are assumed to be constants (we used  $n = 1$ ,  $m = 0.5$ , and  $k \approx 5.5 \cdot 10^{-7} \text{ y}^{-1}$  [2]). The erosion equation captures the river carving effect. Moreover, the uplift  $u$  is uniformly rescaled so that its order of magnitude is close to  $5.0 \cdot 10^{-4} \text{ m y}^{-1}$ , which is a geologically-consistent value (uplift rate in mountain belts rarely exceed a few millimeters per year) that produces visually plausible results.

We extend the original model by coupling the fluvial erosion with a landslide erosion model that mimics the way cliffs are shaped by landslides. This method is derived from the original *thermal erosion* [19] used in Computer Graphics. We check the slope  $s(\mathbf{p})$  after each step of fluvial erosion. If the slope  $s(\mathbf{p})$  is larger than a corresponding talus slope  $\theta$ , we update the height using:

$$\frac{\partial z(\mathbf{p}, t)}{\partial t} = l (s(\mathbf{p}) - \tan \theta),$$

The constant  $l = 10^{-2} \text{ m y}^{-1}$  has been set experimentally and describes the speed of the landslide effect. The talus slope is preset to the commonly accepted default value of  $\theta = 30^\circ$ , but our model enables modifications for each layer of rock, to make it depend on material strength. Considering that the slope at  $\mathbf{p}$  only depends on the height of its lowest neighbor  $z(\mathbf{q})$  and on the horizontal distance between them  $r = \|\mathbf{p} - \mathbf{q}\|$ , we use:

$$s(\mathbf{p}) = \frac{z(\mathbf{p}) - z(\mathbf{q})}{r}.$$

We evaluate  $z$  using an implicit time scheme that is computed in linear time, without the need to invert a matrix: this is achieved by evaluating the heights in the right order, ie.  $z(\mathbf{q}, t + dt)$  before  $z(\mathbf{p}, t + dt)$ .

### 6.2 Rock layers at the surface

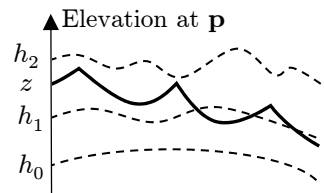


Fig. 11. The nature of the rock at the surface, of altitude  $z$ , is deduced from its position wrt. fold layers of altitudes  $h_0$ ,  $h_1$  and  $h_2$ .

The multi-layered nature of the crust controls not only the folding behavior, but also impacts the erosion process (see Figure 10). An important feature of our subsurface geologically-based simulation is that we can associate a specific rock layer with each visible point of

the eroded terrain surface, and use this information to locally alter the erosional properties or to adapt the rendering.

The rock layer index associated to a point at the eroded surface is obtained from the stratigraphy of the original elevation  $h_0$  of this point inside the crust. This elevation cannot be simply defined as the difference between the total height of uplifted material and the total height of eroded material at that point, because uplift contribution of folds should not be considered when they are being eroded.

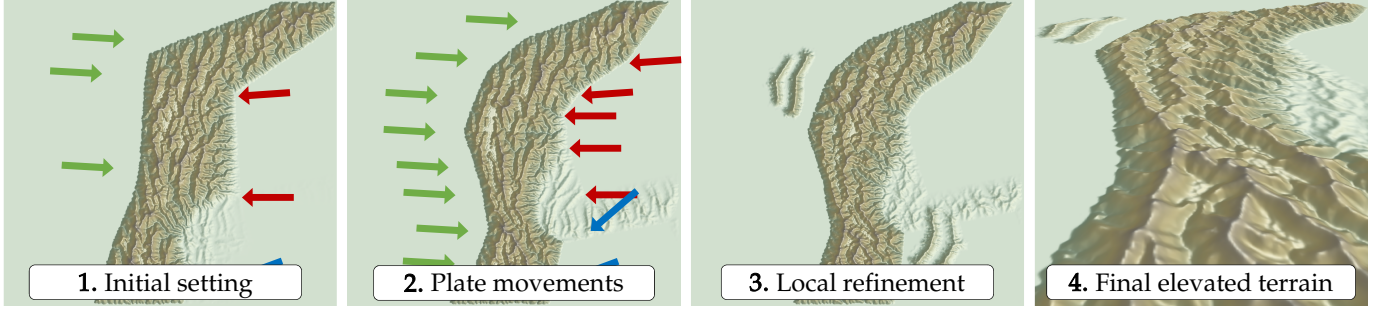


Fig. 12. Successive steps of a sculpting session. The user starts by setting the movements of the plates (1 and 2), before adding local details by choosing folding directions and prescribing specific elevations (3) to get the final result (4).

Therefore, we compute the height of each fold layer at the current simulation step, and evaluate the original height of the surface from its position between the two nearest layers (Figure 11).

The height  $h_k$  of the  $k$ -th layer surface is computed by summing the heights of the layers below, starting with  $h_0 = u_0 t$ , and adding the specific uplift due to folding of the current layer:

$$h_k(\mathbf{p}) = h_{k-1}(\mathbf{p}) + u_k(\mathbf{p}) t,$$

where  $t$  is the current time in the simulation, since the beginning of mountain growth.

We first compute which crustal layer every point of the surface of the terrain  $\mathbf{p}$  belongs to. This is achieved by computing the vertical displacement of each rock layer, from bottom to top, until the total thickness gets higher than the current surface altitude  $z(\mathbf{p})$ ; thus the index  $k$  satisfies  $h_{k-1} < z < h_k$ . We then compute the elevation of the point between the two layers:

$$\gamma = \frac{h_k(\mathbf{p}) - z(\mathbf{p})}{h_k(\mathbf{p}) - h_{k-1}(\mathbf{p})}.$$

The height of the point before deformation is:

$$h_0(\mathbf{p}) = (1 - \gamma) h_k(\mathbf{p}, t_0) + \gamma h_{k-1}(\mathbf{p}, t_0),$$

where  $h_i(\mathbf{p}, t_0)$  denotes the altitude of the crust layer  $i$  before deformation at time  $t_0$ . We derive the layer from the original height of the point. To imitate geological strata, we divide the layers into sub-layers of different strengths, and adjust the erosion parameters accordingly: we use a high erosion constant  $k$  and talus landslide slope ( $\theta = 30^\circ$ ) for weak rock layers, and a low erosion constant  $k$  with a high slope ( $\theta = 50^\circ$ ) for hard rock.

## 7 IMPLEMENTATION, RESULTS, AND DISCUSSION

We developed our algorithm in C++ and with OpenGL and GLSL APIs for real-time rendering. High quality images were directly rendered by streaming our synthesized terrains to *Terragen 3* which was also used to generate terrain surface details. All examples were generated on a desktop computer equipped with an Intel Xeon E5 CPU, clocked at 3.7GHz with 16GB of RAM. We used a Wacom CINTIQ 24HD multitouch table for the input.

### 7.1 Architecture

We separated the user-interaction and the tectonic simulation into two CPU threads. The first thread handles the user-interaction, the visual feedback, and gesture analysis. The viscous material simulation, fold computation and tectonic simulation were performed in the second thread. The separation of tasks into independent threads allowed us to provide authoring in real-time with an average performance over 30Hz.

$\lambda$ [m]	Skeleton edges	Skeletons [ms]	Height [ms]
7,000	955	0.9	2.1
3,000	4,246	3.7	1.4
1,000	36,715	53.0	2.8

TABLE 1

Statistics for the different folds: wavelength (in m), number of skeleton edges, and computation time for the skeleton and the elevation (in ms).

The viscous crust compression is computed on a grid of resolution  $1024 \times 1024$ . The algorithm (Section 4) is straightforward to implement on GPU because each grid cell can be computed independently. We achieve high processing performance (10ms) with no visible grid or aliasing artifacts.

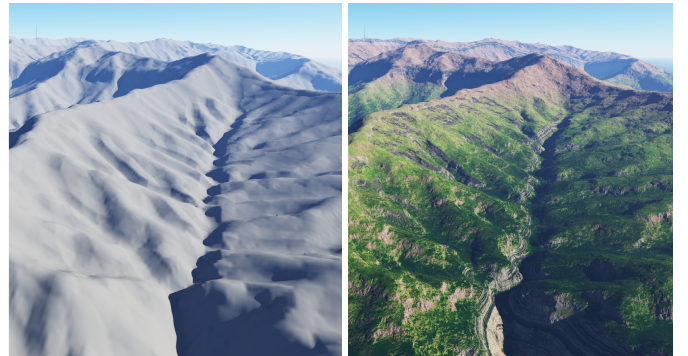


Fig. 13. Real time rendering (left) and the same terrain with details generated by *Terragen* (right).

The folding simulation is divided into two parts: the construction of fold skeletons (Section 5) was implemented on the CPU, whereas the fold height computation using the inverse distance transform was developed on the GPU. Because the distances between the skeleton edges are small and regular, we developed a GPU algorithm that renders

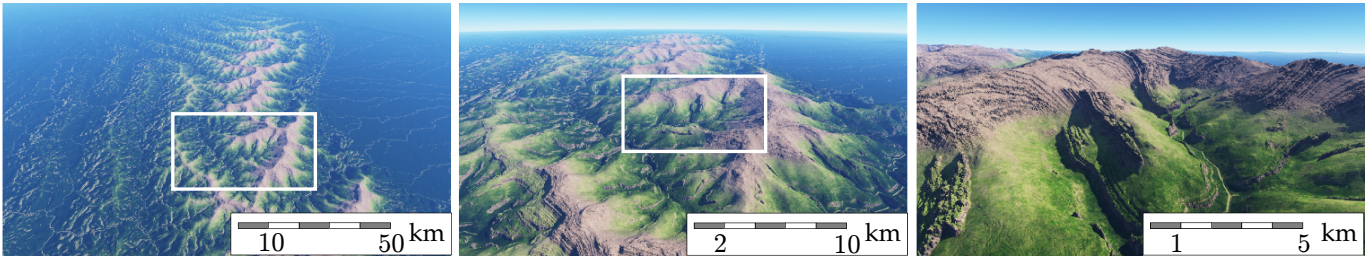


Fig. 14. The user sculpts a mountain range. The results capture the whole range coherency, and massifs are aligned with the fold lines. Eventually, the folds leave marks after erosion.

small rectangles around each line, no larger than the theoretical wavelength, and output the distance to the line at each pixel. The distance at the junction of two lines is defined as the minimum of the results. The performance of this step is detailed in Table 1.

The terrain surface is a mesh with 10,000 vertices for a  $100 \times 100 \text{ km}^2$  terrain. The generation process is fast enough to guarantee interactive visual feedback. One iteration takes 10 ms on average and a 5 million years simulation takes only 5 iterations. Thus, we can perform these iterations interactively before updating the terrain on screen. The parameters used in our setup are reported in Table 2.

$\Delta t$	Total simulation time	$5 \cdot 10^6 \text{ y}$
$n$	Number of layers	2 – 10
$\tau_i$	Layer thickness	0.1 – 5 km
$\mu_i$	Layer viscosity	$10^{10} \text{ Pa}$
$\lambda_i$	Layer fold wavelength	0.5 – 20 km
$\sigma_i$	Layer folding speed	$0.01 \times u - 0.5 \times u$
$u$	Target average uplift	$5.0 \cdot 10^{-4} \text{ m y}^{-1}$
$k$	Erosion constant	$2.5 \cdot 10^{-7} - 8 \cdot 10^{-7} \text{ m}^{-1}$
$\theta_{max}$	Talus angle	$30^\circ - 50^\circ$
$l$	Landslide speed	$10^{-2} \text{ m y}^{-2}$

TABLE 2

Parameters of our system and ranges of values used in the examples.

Only a few post processing steps are needed to achieve high quality rendering. First, the elevation is recomputed off-line by combining the uplifts maps resulting from our method with erosion simulation [25] with a smaller time step and a larger resolution (1,000,000 vertices). A simulation step runs in 4s in average and we need  $\approx 100$  iterations to obtain a final terrain. Finally, we add noise weighted by a function of the slope and the elevation to obtain small surface displacements that are shaded during the final rendering step (Figure 13).

## 7.2 Qualitative and quantitative results

**Sculpting sessions.** Figures 1, 2, 12 and the accompanying video demonstrate the interactive local and global control over the entire process which is one of the important contributions of our method. The user mainly controls his/her design by using hand-on interaction on a multitouch table. By placing and moving both hands on the table, the continental plates are set in motion. The arrows provide visual feedback about the plate shape and location and their rigid

transformations are defined by using finger gestures. The use of the fold refinement tool is illustrated in Figure 12 (3).

**Resulting terrains.** Figures 1, 17, 14 and the accompanying video show high quality renderings. Notice the overall asymmetric range, the alignment of the sub-ranges orthogonal to the compression direction, and at lower scales, the folded strata appearing along eroded cliffs. Figure 20 shows the effects induced by the layered erosion properties on the landforms of the terrain.

In addition to producing still versions of high quality terrains, our method can also be used to produce animations of the formation of mountain ranges, as shown in the accompanying video, in cross section and then as a full rendering.

## 7.3 Validation and discussion

This paper is co-authored by an expert in geology, Jean Braun, who helped us validating the modeling choices and the results. A detailed validation follows:

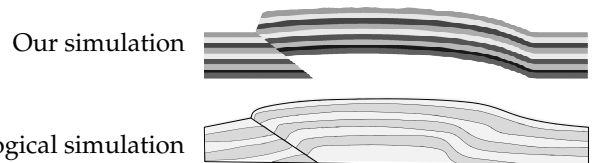


Fig. 15. Comparison between a terrain profile generated by using the compression step only in our method (top) and a geological simulation result, also neglecting folds (bottom, courtesy of J. Braun [44]). Note the similar nearly uniform uplifts in the central parts.

Figure 15 shows a comparison between results from geology [44] and a mountain profile created by our algorithm. It validates the fact that adequately captures the vertical displacement due to crust compression.

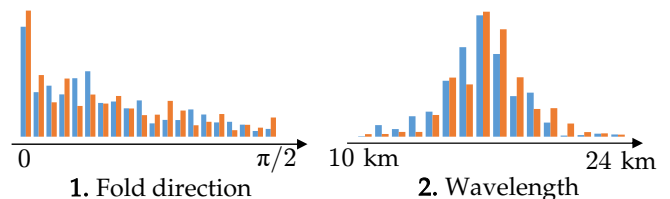


Fig. 16. Folding validation: Comparison of the angles between fold normals and compression direction (left) and the histograms of wavelengths (right) between [45] (blue) and our results (red).

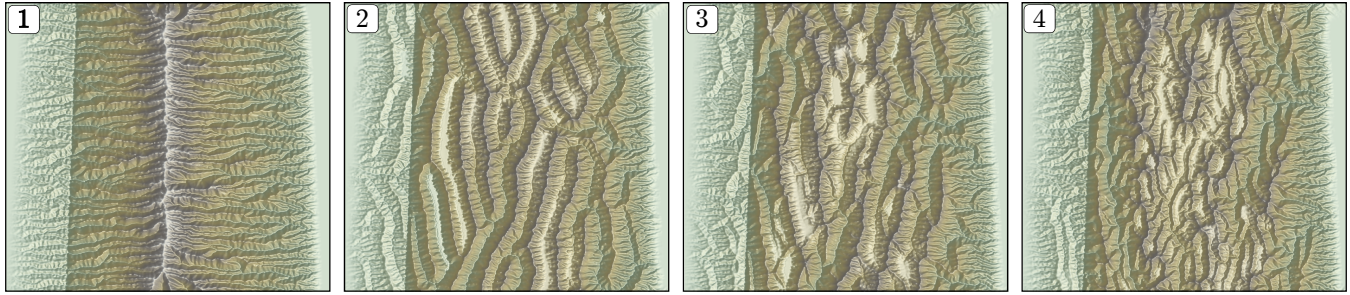


Fig. 17. Various 100 km terrains obtained from the same overall uplift but with different folds layout: only viscous simulation (1), a single layer (2, wavelength  $\lambda = 7$  km), two fold layers (2,  $\lambda_1 = 10$  km,  $\lambda_2 = 5$  km) and (3,  $\lambda_1 = 15$  km,  $\lambda_2 = 3$  km).

Figure 18 shows the folding of one layer in comparison to a geological simulation result [45]. We also provide statistics computed on the graph of these folds. They include wavelength histograms and an histogram of angle between the fold direction and the orthogonal to the compression direction (Figure 16), and a comparison of the graph properties (Table 3). We choose to measure three values that are essential to the visual aspect of the folded regions, namely the number of end nodes, the number of triple joints, and the average fold length (between the nodes or triple joints).

	Our results	Geological simulation [45]
End nodes	21.2 %	25.6 %
Triple linkage	7.5 %	5.6 %
Average fold length	132 km	165 km

TABLE 3  
Comparison of folds graph properties between our results and a geological simulation.

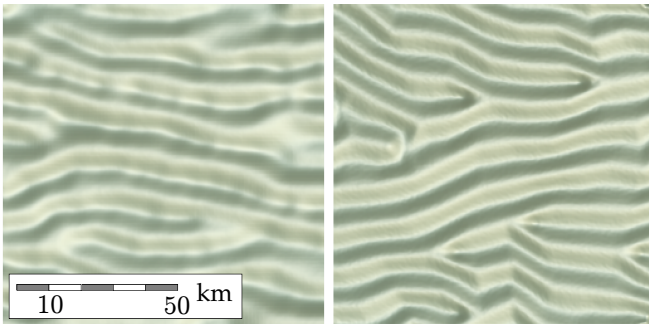


Fig. 18. Folding validation: Comparison between a geological simulation result from [45] (left), and a top view of our results (right), both rendered using the same shading.

We also provide a side-by-side comparison of a result generated by our method with real data from USGS in Figure 19. We selected two mountain ranges that form along plate boundaries, such as the main fault of New Zealand's Alps (Figure 19, top) and the Cerro Chorolque in Southern Bolivia (Figure 19, bottom), i.e. along the eastern side of the Andes. Both formed by thrusting and folding of layered rocks under compression. They differ mostly by their width, with the New Zealand Alps being much narrower than the Andes. Deformation in the New Zealand Alps is focused on a single crustal-scale fault (the Alpine Fault) and built out of a single lithology, the Otago Schist, whereas in the Andes, the stratigraphy is more complex and controls the

spacing between the crustal-scale faults and folds. The main difference between real data and our results is caused by glacial erosion, which is not captured by our method, and results in broader, flat-bottom (or U-shaped) valleys.

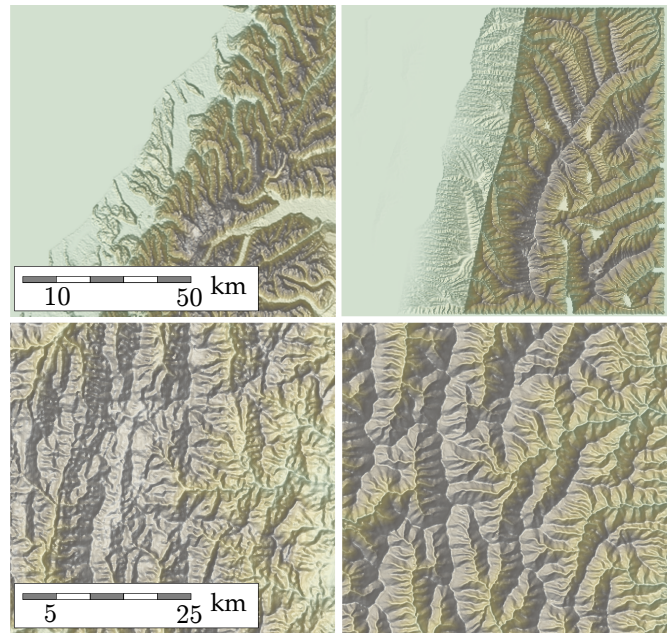


Fig. 19. Comparisons between real mountains (left) and our results (right). Top: we compare the main fault with New Zealand's Alps. Bottom: we focus on eroded folds in the Andes.

The importance of including folding and a stratigraphic control on erodibility is evidenced in our examples. Where such folding and stratigraphic control is not included (as shown in Figure 17.1), all valleys strike perpendicular to the axis of the mountain. Although such an organization of drainage can be observed locally, the morphology of most mountain ranges at the largest scale is strongly controlled by crustal-scale folding. In some mountain ranges, folding is mainly controlled by a single layer (Figure 17.2), such as in the Zagros Mountains in Iran which formed by folding of a sedimentary sequence above a weak salt layer. Various results can be obtained by changing the relative size of each fold layer: Figure 17.3 includes two layers of related fold wavelength (5 and 10 km), and can be compared to the real data from the Andes (Figure 19, bottom), while Figure 17.4 shows a mountain formed from a larger difference in fold wavelength (3 and 15 km). At the smaller scale,

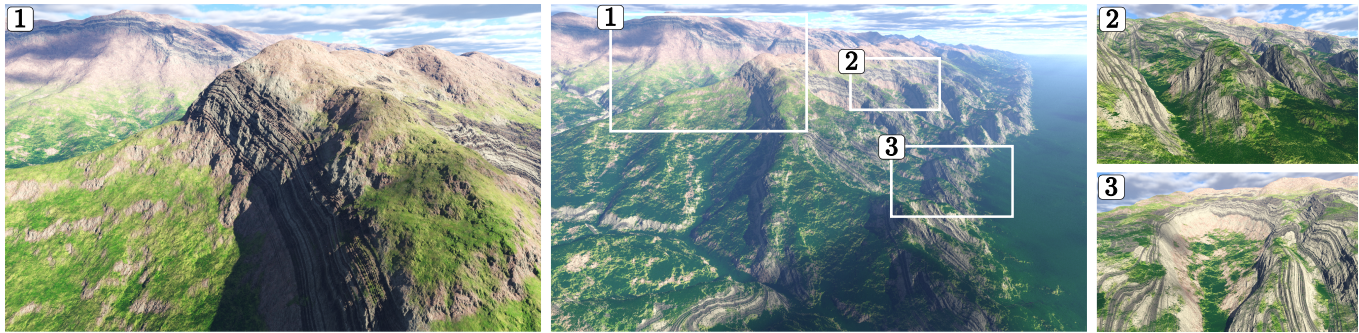


Fig. 20. The difference of erosion properties among rock layers induces different effects: cliffs (1), individual small summits (2), or cirque (3).

most mountainous landforms are also strongly controlled by the progressive exhumation of rock layers of varying resistance to erosion (Figure 20). A dramatic example of this, that should be compared to our simulated landscape shown in Figures 19.2 and 20.3, is the so-called *velodrome* structure near the city of Digne in the French Alps (Figure 19.1) which formed by erosion of a gently folded stack of sedimentary rocks of alternating resistance to erosion.

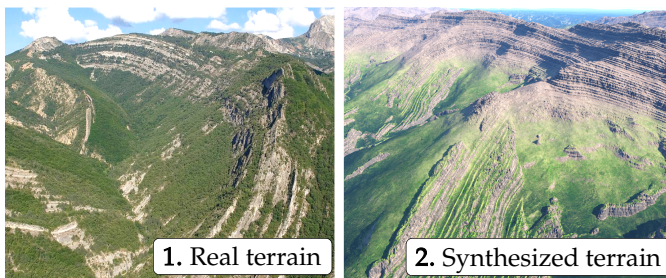


Fig. 21. Comparison of a photograph showing real folds (1) and our synthesized terrain (2).

**Limitations** Our sculpting method has several limitations. First, the fold model we use is only valid at early folding stages: when a real mountain fold grows too high, the relation with wavelength used in this paper is no longer true, and the fold does not have a sinusoid shape anymore [38]. A volumetric simulation would be necessary to compute more complex behaviors such as fold distortion [46], as well as secondary faults within the crust. While such a simulation would be interesting and would capture a wider range of geological situations in the generated folds, it would be also difficult to control.

Second, we do not allow for a plate setup that evolves over time, though it is the case in several real mountain ranges. This would be more geologically correct, but it would provide less intuitive manipulation.

Finally, our method does not lend itself for precise editing as it only provides the user with a high level of control over the final result. Although this limitation is alleviated by the possibility to edit fold skeletons, the user cannot author certain details, such as the number of dendritic patterns while keeping the other unchanged as shown in Figure 20.

## 7.4 User Study

We conducted a user study to evaluate the usability of our mountain sculpting tool. It was tested on 11 subjects who

had low to medium knowledge of geology. Among them, three were professional CG artists. The participants first performed an informal learning trial in order to understand the tool. After that, they were asked to reproduce real mountain ranges shown in two images: the Caucasus and the Alps. The users had to evaluate the interactivity and the effectiveness of the method, and the quality of their results. The measurements were rated on a 1-4 Likert's scale.

The quantitative part of the user study showed that the participants had a positive experience while working with our sculpting tool. The effectiveness of the tectonic gestures averaged 3.3, the local control tool received a mark of 2.8 and the overall intuitiveness was 3.0. The users rated their own achievement in reproducing the shape of real ranges with an average of 2.6.

The qualitative part of the user study included some positive comments from the subjects regarding the simplicity of the first contact, the enabled creativity, and the realism of the results. An important feature of the editing tool was the use of arrows to symbolize tectonic motion. The automatic update of arrows during interaction so that they always show the same rigid transformation, was pointed out by users: some found this counterintuitive, whereas most found it to be of great help. The interactivity and simplicity of our system has potential applications in education, in particular for teaching geology: in the field, the geometry of a fold may be partly obscured by the way erosion has revealed it to the geologist. Our system can show the fold as texture on a cliff wall after erosion as well as its true 3D geometry, making it a useful tool for training geologists in reconstructing the fold 3D geometry from its exposed geometry.

## 8 CONCLUSION

We have presented the first approach enabling users to progressively sculpt large-scale mountain ranges by using hands-on interaction with displacement gestures, as if they are sculpting plasticine. We use two coupled sub-models, one for modeling the local thickening of the earth's crust due to compression of colliding plates, and the other for modeling multi-scale folding behavior. Interleaving these two sub-models results in a complex uplift map simulating the growth rate of mountain ranges between tectonic plates. This enables us to generate visually compelling terrains at interactive rates.

Independent of the interaction, the terrain the user manipulates evolves according to first-order geological laws.

Consequently, our validations show that terrains modeled using our system capture important landform features of mountain ranges, such as the asymmetry in landform resulting from the presence of a crustal scale fault and multi-scale folding of crust layers appearing in eroded regions.

In future work, we would like to take into account other types of tectonic activity such as volcanoes, and other types of erosion phenomena such as glacial erosion. In addition to the application of our sculpting system to interactive terrain design, we would like to apply them as an educational tool for teaching geology. Lastly, the user input could be complemented by a time-line allowing to create easily tunable animations of the formation of mountain ranges.

## APPENDIX

### A-GLOSSARY

**Tectonic plates:** large pieces of the outermost layer of the Earth that moves as coherent plates driven by convective motions in the underlying mantle. Two types of plate exist: continental plates which have a crust made of relatively lighter granitic rocks and oceanic plates which have a crust made of relatively heavier basaltic rocks. The motion of the plates is facilitated by the presence of a low viscosity layer just beneath the plates called the asthenosphere. On geological time scales, the asthenosphere appears as a quasi inviscid fluid.

**Crust:** the crust is the upper most layer of the Earth; it is chemically different from the underlying mantle. Its thickness varies between 30-40 km in continental areas and 5-10 km in oceanic areas.

**Isostasy:** according to the principle of isostasy, the tectonic plate can be regarded as "floating" at the Earth's surface on a low viscosity denser "fluid" called the asthenosphere. Due to its low viscosity, the asthenosphere cannot sustain large stress gradient. Any variations in crustal thickness must be accompanied by vertical motion of the surface such that the weight of any column of rock above the asthenosphere remains constant on geological time scales. This explains why mountains are regions of thickened crust while basins are characterized by an unusually thin crust.

**Continental collision:** as tectonic plates move at the Earth's surface they interact with each other. Where their relative motion is convergent, a mountain belt is created such as in the European Alps; where their relative motion is divergent a rift develops, such as in East Africa. A few plate boundaries are characterized by a strike-slip or mostly oblique relative motion, such as along the San Andreas Fault system.

**Stratigraphy:** term that is commonly used to describe the near-horizontal layering of rocks in the crust that is created by the process of deposition or sedimentation. The deep crust is also compositionally layered with quartz-rich rocks near the surface and feldspar-rich rocks near its base.

**Competence:** resistance of rocks to deformation or erosion due to its relative mechanical strength.

**Erosion:** process by which the shape of the Earth's surface is altered by transport of rocks, most commonly from regions of high elevation to regions of low elevation. This erosion is performed by flowing rivers and sliding glaciers, but

also by more local processes, such as soil creep, landsliding, chemical dissolution or eolian erosion.

**Fault and faulting:** a discontinuity in the Earth's crust that accommodates motion between two adjacent crustal blocks. Faulting takes place at a large variety of scales (from a few mm to hundreds of km).

**Fold and folding:** mechanical instability that arises when a strong layer embedded in a weaker material is shortened leading to a quasi sinusoidal pattern of deformation. Folds are commonly observed in geological materials, mostly due to the existence of a horizontal strong layering in the Earth's crust.

### ACKNOWLEDGMENTS

This work was partially funded by the advanced grant #291184 EXPRESSIVE from the European Research Council (ERC-2011-ADG\_20110209) and by the NSF grant #1606396 *Haptic-Based Learning Experiences as Cognitive Mediators for Conceptual Understanding and Representational Fluency in Engineering Education*. We also thank Intel and Nvidia for providing computer hardware.

### REFERENCES

- [1] S. Willett, C. Beaumont, and P. Fullsack, "Mechanical model for the tectonics of doubly vergent compressional orogens," *Geology*, vol. 21, no. 4, pp. 371-374, 1993.
- [2] J. Braun and S. D. Willett, "A very efficient  $O(n)$ , implicit and parallel method to solve the stream power equation governing fluvial incision and landscape evolution," *Geomorphology*, vol. 180, pp. 170-179, 2013.
- [3] J. Braun and P. Yamato, "Structural evolution of a three-dimensional, finite-width crustal wedge," *Tectonophysics*, vol. 484, pp. 181-192, 2010.
- [4] B. Benes and R. Forsbach, "Layered data representation for visual simulation of terrain erosion," in *Proceedings of SCCG*, vol. 25(4). IEEE Computer Society, 2001, pp. 80-86.
- [5] S. Ellis, "Forces driving continental collision: reconciling indentation and mantle subduction tectonics," *Geology*, vol. 24, no. 4, pp. 699-702, 1996.
- [6] M. Natali, E. Lidal, J. Parulek, I. Viola, and D. Patel, "Modeling terrains and subsurface geology," *Proc. of Eurogr. State of the Art Reports*, pp. 155-173, 2013.
- [7] R. M. Smelik, T. Tutenel, R. Bidarra, and B. Benes, "A survey on procedural modelling for virtual worlds," *Comp. Graph. Forum*, vol. 33, no. 6, pp. 31-50, 2014.
- [8] D. S. Ebert, F. K. Musgrave, D. Peachey, K. Perlin, and S. Worley, *Texturing and Modeling: A Procedural Approach*. Morgan Kaufmann Publishers Inc., 2002.
- [9] P. Prusinkiewicz and M. Hammel, "A fractal model of mountains with rivers," in *Proc. of Graph. Interface*, vol. 30(4), 1993, pp. 174-180.
- [10] S. Stachniak and W. Stuerzlinger, "An algorithm for automated fractal terrain deformation," in *Proceedings of Computer Graphics and Artificial Intelligence*, vol. 1, 2005, pp. 64-76.
- [11] A. D. Kelley, M. C. Malin, and G. M. Nielson, "Terrain simulation using a model of stream erosion," *ACM Trans. Graph.*, pp. 263-268, 1988.
- [12] H. Hnaidi, E. Guérin, S. Akkouché, A. Peytavie, and E. Galin, "Feature based terrain generation using diffusion equation," *Comp. Graph. Forum*, vol. 29, no. 7, pp. 2179-2186, 2010.
- [13] J.-D. G enevaux, E. Galin, E. Gu erin, A. Peytavie, and B. Benes, "Terrain generation using procedural models based on hydrology," *ACM Trans. Graph.*, vol. 32, no. 4, pp. 143:1-143:13, 2013.
- [14] J.-D. G enevaux, E. Galin, A. Peytavie, E. Gu erin, C. Briquet, F. Grosbellet, and B. Benes, "Terrain modelling from feature primitives," *Comp. Graph. Forum*, vol. 34, no. 6, pp. 198-210, 2015.
- [15] E. Gu erin, J. Digne, A. Peytavie, and E. Galin, "Sparse representation of terrains for procedural modeling," *Comp. Graph. Forum*, 2016.

- [16] H. Zhou, J. Sun, G. Turk, and J. M. Rehg, "Terrain synthesis from digital elevation models," *IEEE TVCG*, vol. 13, no. 4, pp. 834–848, 2007.
- [17] J. Gain, B. Merry, and P. Marais, "Parallel, realistic and controllable terrain synthesis," *Comp. Graph. Forum*, vol. 34, no. 2, pp. 105–116, 2015.
- [18] F. P. Tasse, A. Emilien, M.-P. Cani, S. Hahmann, and A. Bernhardt, "First person sketch-based terrain editing," in *Proceedings of the Graphics Interface*, 2014, pp. 217–224.
- [19] F. K. Musgrave, C. E. Kolb, and R. S. Mace, "The synthesis and rendering of eroded fractal terrains," *ACM SIGGRAPH Computer Graphics*, vol. 23, no. 3, pp. 41–50, 1989.
- [20] B. Neidhold, M. Wacker, and O. Deussen, "Interactive physically based Fluid and Erosion Simulation," in *Eurographics Workshop on Nat. Phen.*, 2005.
- [21] B. Benes, V. Těšínský, J. Hornýš, and S. K. Bhatia, "Hydraulic erosion," *Comp. Anim. and Virt. Worlds*, vol. 17, no. 2, pp. 99–108, 2006.
- [22] X. Mei, P. Decaudin, and B.-G. Hu, "Fast hydraulic erosion simulation and visualization on gpu," in *Pacific Graphics*, vol. 0. IEEE Computer Society, 2007, pp. 47–56.
- [23] J. Vanek, B. Benes, A. Herout, and O. Stava, "Large-scale physics-based terrain editing using adaptive tiles on the GPU," *Comp. Graph. and App., IEEE*, vol. 31, no. 6, pp. 35–44, 2011.
- [24] S. Reed, O. Kreylos, S. Hsi, L. Kellogg, G. Schladow, M. Yikilmaz, H. Segale, J. Silverman, S. Yalowitz, and E. Sato, "Shaping watersheds exhibit: An interactive, augmented reality sandbox for advancing earth science education," in *AGU Fall Meeting Abstracts*, vol. 1, 2014, p. 01.
- [25] G. Cordonnier, J. Braun, M.-P. Cani, B. Benes, E. Galin, A. Peytavie, and G. Eric, "Large Scale Terrain Generation from Tectonic Uplift and Fluvial Erosion," *Comp. Graph. Forum*, 2016.
- [26] E. Michel, A. Emilien, and M.-P. Cani, "Generation of Folded Terrains from Simple Vector Maps," in *Eurographics 2015 short paper proceedings*, Zurich, Switzerland, May 2015, p. 4.
- [27] T. A. Galyean and J. F. Hughes, "Sculpting: An interactive volumetric modeling technique," in *Computer Graphics (Proceedings of SIGGRAPH 91)*, vol. 25, no. 4, Jul. 1991, pp. 267–274.
- [28] G. Dewaele and M.-P. Cani, "Interactive global and Local Deformations for Virtual Clay," *Graphical Models*, vol. 66, no. 6, pp. 352–369, 2004, (from Pacific Graphics'2003). [Online]. Available: <https://hal.inria.fr/inria-00537468>
- [29] O. Sorkine and M. Alexa, "As-rigid-as-possible surface modeling," in *Symposium on Geometry processing*, vol. 4, 2007.
- [30] L. Stanculescu, R. Chaine, M.-P. Cani, and K. Singh, "Sculpting multi-dimensional nested structures," *Comp. and Graphics*, vol. 37, no. 6, pp. 753–763, 2013. [Online]. Available: <https://hal.inria.fr/hal-00865552>
- [31] A. Milliez, M. Wand, M.-P. Cani, and H.-P. Seidel, "Mutable elastic models for sculpting structured shapes," *Comp. Graph. Forum*, vol. 32, no. 2pt1, pp. 21–30, 2013.
- [32] A. Peytavie, E. Galin, J. Grosjean, and S. Mérillou, "Arches: a framework for modeling complex terrains," *Comp. Graph. Forum*, vol. 28, no. 2, pp. 457–467, 2009.
- [33] D. S. Lopes, D. Mendes, M. Sousa, and J. Jorge, "Expeditious illustration of layer-cake models on and above a tactile surface," *Computers & Geosciences*, vol. 90, pp. 1–9, 2016.
- [34] M. Natali, J. Parulek, and D. Patel, "Rapid modelling of interactive geological illustrations with faults and compaction," in *Proceedings of the 30th Spring Conference on Computer Graphics*. ACM, 2014, pp. 5–12.
- [35] S. Castellort, L. Goren, S. Willett, J.-D. Champagnac, F. Herman, and J. Braun, "River drainage patterns in the New Zealand Alps primarily controlled by plate tectonic strain," *Nature Geoscience*, vol. 5, pp. 744–748, 2012.
- [36] P. Yamato, B. J. Kaus, F. Mouthereau, and S. Castellort, "Dynamic constraints on the crustal-scale rheology of the zagros fold belt, iran," *Geology*, vol. 39, no. 9, pp. 815–818, 2011.
- [37] O. Rémillard and P. G. Kry, "Embedded thin shells for wrinkle simulation," *ACM Trans. Graph.*, vol. 32, 2013.
- [38] S. M. Schmalholz and Y. Y. Podladchikov, "Finite amplitude folding: Transition from exponential to layer length controlled growth," *Earth and Planetary Sci. Letters*, vol. 179, pp. 363–377, 2000.
- [39] R. Narain, A. Samii, and J. F. O'Brien, "Adaptive anisotropic remeshing for cloth simulation," *ACM Trans. Graph.*, vol. 31, no. 6, p. 152, 2012.
- [40] D. Rohmer, T. Popa, M.-P. Cani, S. Hahmann, and A. Sheffer, "Animation wrinkling: augmenting coarse cloth simulations with realistic-looking wrinkles," in *ACM Trans. Graph.*, vol. 29, no. 6. ACM, 2010, p. 157.
- [41] M. A. Biot, "Theory of folding of stratified viscoelastic media and its implications in tectonics and orogenesis," *Geological Society of America Bulletin*, vol. 72, no. 11, pp. 1595–1620, 1961.
- [42] J. Braun, D. Zwart, and J. Tomkin, "A new surface-processes model combining glacial and fluvial erosion," *Annals of Glaciology*, vol. 28, pp. 282–290, 1999.
- [43] A. D. Howard, "Equilibrium and time scales in geomorphology: Application to sand-bed alluvial streams," *Earth Surface Processes and Landforms*, vol. 7, no. 4, pp. 303–325, 1982.
- [44] J. Braun and M. Sambridge, "Dynamical lagrangian remeshing (dlr): a new algorithm for solving large strain deformation problems and its application to fault-propagation folding," *Earth and Planetary Sci. Letters*, vol. 124, no. 1, pp. 211–220, 1994.
- [45] N. Fernandez and B. J. Kaus, "Fold interaction and wavelength selection in 3d models of multilayer detachment folding," *Tectonophysics*, vol. 632, pp. 199–217, 2014.
- [46] B. J. Kaus and S. M. Schmalholz, "3d finite amplitude folding: implications for stress evolution during crustal and lithospheric deformation," *Geophysical Research Letters*, vol. 33, no. 14, 2006.



**Guillaume Cordonnier** is a former student of the Ecole Normale Supérieure de Lyon, and obtained his master degree from Institut Polytechnique de Grenoble. He is pursuing his PhD degree at Université Grenoble Alpes on geology based terrain modeling. His research interests cover procedural modeling and simulation of natural phenomena.



**Marie-Paule Cani** is a Professor of Computer Science at Grenoble University and Inria. Her research interests cover Shape Modeling and Computer Animation. She contributed to a number of high level models for shapes and motion such as implicit surfaces, multi-resolution physically-based animation and hybrid representations for real-time natural scenes.



**Bedrich Benes** is a professor of Computer Graphics Technology at Purdue University and a director of the High Performance Computer Graphics Laboratory. His area of research is in procedural and inverse procedural modeling and simulation of natural phenomena and he has published over 100 research papers in the field.



**Jean Braun** is Senior Scientist at the GFZ German Research Center for the Geosciences in Potsdam, Germany and Professor at the University of Potsdam. His research interests include the quantification of processes responsible for the evolution of the Earth's surface on geological time scales.



**Eric Galin** is a Professor of Computer Science at the University Lumière Lyon 2 and researcher at the LIRIS laboratory. His research interests in computer graphics focus on procedural modeling of virtual worlds, inverse procedural modeling, simulating natural phenomena and implicit surface modeling.