Fractal approximation of surfaces based on projected IFS attractors

E. Guérin, E. Tosan and A. Baskurt

LIGIM - EA 1899 - Computer Graphics, Image and Modeling Laboratory
Claude Bernard University, Lyon, France
[eguerin | et | abaskurt]@ligim.univ-lyon1.fr

Abstract

A method for approximating smooth or rough surfaces defined in $\mathbb{R}^3$ is introduced. A fractal model called projected IFS model allows the extension of the iteration space to a barycentric space $\mathbb{R}^{n^2}$ by enriching the classical IFS model with a set of control points (m^2 points). This flexible model has good fitting properties for recovering surfaces. The input for the model is single viewpoint range data defined on a fixed grid and also 2D grey-level images considered as surfaces. The model recovery is formulated as a non-linear fitting problem and resolved using a modified LEVENBERG-MARQUARDT *minimization method. During the iterative fitting algorithm, all the parameters of the projected IFS model are adjusted simultaneously in order to minimize the overall distance between the model's surface and the original data. The final model is very compact and gives satisfactory results on synthetic range data and real geological surfaces. The main applications are surface modeling, shape description and geometric surface compression.*

1. Introduction

Basically, the problem of approximating the surface of 3D objects consists in finding a model that represents a set of data points:

$$(x_i, y_i, z_i) \in \mathbb{R}^3, \ \forall i = 0, \ldots, n$$

A wide variety of representation methods have been proposed for modeling these surfaces[1]. Basically, they can be classified into three categories depending on the data source and the target application: mesh representation, parametric representation and implicit representation. For a simple visualization of smooth surfaces, the model widely used is the mesh approximation[2]. When the data is issued from sensors and the model should be suitable to a CAD use, parametric approximation seems to be well adapted using a standard model such as NURBS or B-splines[2]. If the approximation should be used for a more semantic description of an object, implicit models can be chosen like superquadrics[3] [4]. Each application domain has a preferred model that relies on its specificities. Unfortunately, these models do not recover rough surfaces, i.e. surfaces defined by continuous functions that are nowhere differentiable.

In order to propose an efficient solution to the problem of rough surface approximation, the current study proposes a parametric model based on a fractal model. In [5] and [6], we have proposed a model for fractal curve and surfaces. This model combines the classical Iterated Function System (IFS) model and the free form approximation theory based on a set of control points. These points allow an easy and flexible control of the fractal shape generated by the IFS model and provide a high quality fitting, even for surfaces with sharp transitions. This model is called the projected IFS model. In [7] and [8], we have proposed an approximation method for curves based on this model. In this paper, we give the extension of this method to surfaces.

2. Approximation model

2.1. Iterated Function Systems (IFS)

Introduced by BARNSLEY[9] in 1988, the IFS model generates a geometrical shape or an image [10] with an iterative process. An IFS-based modeling system is defined by a triple $(\mathcal{X}, d, \mathcal{S})$ where:

- $(\mathcal{X}, d)$ is a complete metric space, $\mathcal{X}$ is called *iteration space*;

- $\mathcal{S}$ is a semigroup acting on points of $\mathcal{X}$ such that: $p \mapsto Tp$ where T is a contractive operator, $\mathcal{S}$ is called *iteration semigroup*.

An *IFS* $\mathcal{I}$ *(Iterative Function System)* is a finite subset of $\mathcal{S}$: $\mathcal{I} = \{T_0, ..., T_{N-1}\}$ with operators $T_i \in \mathcal{S}$. We note $\mathcal{H}(\mathcal{X})$ the set of non-empty compacts of $\mathcal{X}$. The associated HUTCHINSON operator is:

$$K \in \mathcal{H}(\mathcal{X}) \quad \mapsto \quad \mathcal{I}K = T_0K \cup ... \cup T_{N-1}K$$

This operator is contractive in the new complete metric space $\mathcal{H}(\mathcal{X})$ and admits a fixed point, called *attractor* [9]:

$$\mathcal{A}(\mathcal{I}) = \lim_{n\to\infty} \mathcal{I}^n K \; with \; K \in \mathcal{H}(\mathcal{X})$$

When operators match joining conditions[11], this attractor is a surface:

$$\mathcal{A}(\mathcal{I}) = \left\{ \phi(s,t) \,|\, (s,t) \in [0,1]^2 \right\}$$

where ϕ is a function $\phi : [0,1]^2 \to \mathcal{X}$.

2.2. Projected IFS attractors

In classical fractal interpolation [9] or fractal compression [10], the complete metric space $\mathcal{X}$ used is $\mathbb{R}$ or $\mathbb{R}^2$, and the iteration semigroup is constituted of contractive affine operators. Our work consists in enlarging iteration spaces [5] [6]. This model uses a barycentric space $\mathcal{X} = \mathcal{B}^J$:

$$\mathcal{B}^J = \{(\lambda_j)_{j\in J} \,|\, \sum_{j\in J} \lambda_j = 1\}$$

For surfaces, this barycentric space is used with:

$$J = \{0, \ldots, m\} \times \{0, \ldots, m\}$$

Then, the iteration semigroup is constituted of matrices with barycentric columns:

$$S_J = \{T \,|\, \sum_{j\in J} T_{ij} = 1, \forall i \in J\}$$

This choice leads to the generalization of IFS attractors named **projected IFS attractors**:

$$P\mathcal{A}(\mathcal{I}) = \{P\lambda \,|\, \lambda \in \mathcal{A}(\mathcal{I})\}$$

where P is a grid of control points:

$$P = (p_j)_{j\in J}$$

and $P\lambda = \sum_{j\in J} \lambda_j p_j$. In this way, we can construct a fractal function [6] [11] using the projection:

$$F(s,t) = P\phi(s,t) = \sum_{j\in J} \phi_j(s,t) p_j$$

where $\phi(s,t)$ is a vector of functions:

$$\phi(s,t) = (\phi_j(s,t))_{j\in J}$$

and $J = \{0, \ldots, m\} \times \{0, \ldots, m\}$.

2.3. Parametric representation of the surface

The model allows to calculate theoritically the exact value $\phi(s,t)$ for any value of the parameters s and t. This means that we are able to recover the surface value $F(s,t) = P\phi(s,t)$. Let consider the number of transformations N of $\mathcal{I}$ is b^2. To each value of s and t can be associated a development $\alpha(s)$ and $\alpha(t)$ in a b-base such that[12]:

$$s = \sum_{i=1}^{\infty} \frac{1}{b^i} \alpha_i(s)$$

where $\alpha_i(s) = 0 \ldots (b-1)$.

Classically, we use the PEANO combination of the development of $\alpha(s)$ and $\alpha(t)$. Here is an example of a 3×3 subdivision scheme of square with the PEANO indexing of the transformations:

T_2	T_5	T_8
T_1	T_4	T_7
T_0	T_3	T_6

Let us denote $\beta_i(s,t) = \alpha_i(s) \natural \alpha_i(t)$ this index for notation simplifications:

$$\begin{array}{rcl} \beta : [0,1]^2 & \to & \{0, \ldots, b^2-1\}^\omega \\ (s,t) & \mapsto & \beta_1(s,t) \ldots \beta_n(s,t) \ldots \end{array}$$

where $\natural$ is the following mapping:

$$\natural : \{0, \ldots, b-1\} \times \{0, \ldots, b-1\} \to \{0, \ldots, b^2-1\}$$

and $\{0, \ldots, b^2-1\}^\omega$ is the set of infinite words of $\{0, \ldots, b^2-1\}$.

Then, the value of $\phi(s,t)$ is computed iteratively[9]:

$$\phi(s,t) = \lim_{k\to\infty} T_{\alpha_1(s)\natural\alpha_1(t)} \ldots T_{\alpha_k(s)\natural\alpha_k(t)} \lambda \;\; \forall \lambda \in \mathcal{X}$$

Note that $\beta_i(s,t)$ is the index of the operator T applied on the i^{th} iteration in the process of recovering $\phi(s,t)$.

In this way, we can construct a fractal parametric surface characterized by a grid of control points P using the projection. Fig. 4 and 5 show two surfaces generated with the same IFS model using two different 4×4 control grids.

2.4. Tabulation of the parametric surface

With a tabulation process, considering only the values of s and t multiple of $\frac{1}{b^p}$ leads to a simplification in the computing without any loss of information. The surface tabulation is a grid defined by:

$$F(\frac{i}{b^p}, \frac{j}{b^p}) = P \lim_{k\to\infty} T_{\beta_1(\frac{i}{b^p}, \frac{j}{b^p})} \ldots T_{\beta_k(\frac{i}{b^p}, \frac{j}{b^p})} \lambda, \; \forall \lambda \in \mathcal{B}^J$$

We know the development of $\frac{i}{b^p}$:
For $i = 0, \ldots, b^p - 1$:

$$\alpha(\frac{i}{b^p}) = \alpha_1(\frac{i}{b^p}) \ldots \alpha_p(\frac{i}{b^p}) 0 0 \ldots$$

Then, $F(\frac{i}{b^p},\frac{j}{b^p})$ simplifies in[12]:

$$F(\frac{i}{b^p},\frac{j}{b^p}) = PT_{\beta_1(\frac{i}{b^p},\frac{j}{b^p})}\dots T_{\beta_p(\frac{i}{b^p},\frac{j}{b^p})}\,\phi(0,0)$$

We choose $\phi(0,0) = e_{00}$ where e_{00} is the first base vector of $\mathcal{B}^J$:

$$e_{00} = (1,0,\dots,0)^T$$

In this way, the surface tabulation can be generated computing only p iterations without any loss of information:

$$F(\frac{i}{b^p},\frac{j}{b^p}) = PT_{\beta_1(\frac{i}{b^p},\frac{j}{b^p})}\dots T_{\beta_p(\frac{i}{b^p},\frac{j}{b^p})}\,e_{00}$$

$\forall (i,j) \in 0,\dots,(b^p-1)^2$.

Note that for $i = b^p$ or/and $j = b^p$ the development can not be expressed as a word that ends with an infinite suite of zeros, but with an infinite suite of $(b-1)$. Therefore, we have to make other choices such as $\phi(0,1) = e_{0(b-1)}$ in order to simplify expressions.

3. Approximation method

Given a sampled surface $(s_i,t_i,\mathbf{Q}_{ij}) \in \mathbb{R}^3$, the challenge is to determine the projected IFS model which provides a high quality approximation of this surface. The approach proposed in the current study is similar to the one we introduced in [7] [8] for curves. It is based on a non-linear fitting formalism. In this section, the way to construct a fractal surface family is detailed. Then, we show how the approximation problem can be seen as a standard non-linear fitting problem. Finally, we propose a resolution based on the LEVENBERG-MARQUARDT algorithm.

3.1. Projected IFS for surface modeling

In order to illustrate our method and also to provide a compact descriptor for the shape, some restrictions are applied to the model. IFS transformations are reduced to a 3×3 square subdivision, i.e. $b = 3$. Control points constitute a 4×4 grid, i.e. $J = \{0,1,2,3\}\times\{0,1,2,3\}$.

The fractal function is a grey-level function defined by a 4×4 grid $P = (z_j)_{j\in J}$ of control scalar values:

$$(s,t) \in [0,1]^2 \mapsto F(s,t) = \sum_{j\in J} \phi_j(s,t) z_j \in \mathbb{R}$$

Each transformation T_i acts locally on a subset of control points. It means that $(T_i)_{kl} \neq 0$ only for a few values of k and l depending of the transformation T_i considered. Furthermore, joining conditions of IFS $\mathcal{I} = \{T_0,\dots,T_{N-1}\}$ imply identification of columns:

$$(T_i)\,e_j = (T_{i'})\,e_{j'}$$

Then, the IFS is described by a subdivision scheme. Fig. 1 summarizes this local subdivision scheme. It also details how each control point is transformed depending on its spatial position on the grid. Fig. 2 shows three iterations of the construction process. Fig. 3 shows the same process presenting the surfaces as 2D grey-level images.

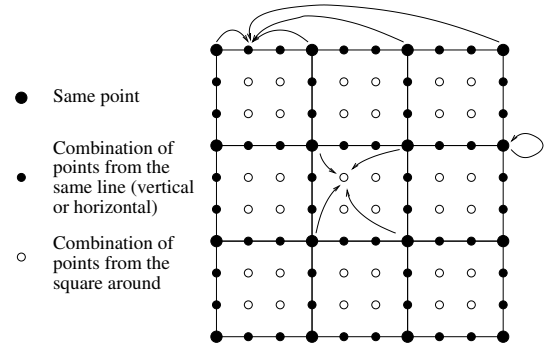


Figure 1: *Subdivision scheme based on a* 4×4 *control grid.*

3.2. Surface family

Considering the coefficients of the operators T_j and the coordinates of the control grid P as elements of a parameter vector $\mathbf{a}$ allows us to construct a fractal surface family. Thanks to the simplifications detailed in the previous section, only a few number of parameters is needed for representing the projected IFS model:

- $4\times 4 = 16$ for the control grid (containing only the z coordinates).
- $4-1 = 3$ coefficients per new point defined as a combination of 4 aligned points (small points in Fig. 1). There is a total of 36 points in this category.
- $4-1 = 3$ coefficients per new point defined as a combination of 4 points around (small circles in Fig. 1). There is also a total of 36 points in this category.

Then, a vector $\mathbf{a}$ of 232 parameters is constructed to code the fractal model. Let $F_{\mathbf{a}}$ be the surface generated with the model described with the parameter vector $\mathbf{a}$.

3.3. Non linear fitting

Let $\mathbf{Q}_{ij\,(i=0,\dots,b^p\ j=0,\dots,b^p)}$ be a given surface to approximate. The approximation problem consists in determining the parameter vector $\mathbf{a}$ that minimizes the distance between the sampled surface $\mathbf{Q} = \{(\frac{i}{b^p},\frac{j}{b^p},\mathbf{Q}_{ij})\}$ and the function $F_{\mathbf{a}}$:

$$\mathbf{a}_{opt} = \underset{\mathbf{a}}{\operatorname{argmin}}\ d(\mathbf{Q},F_{\mathbf{a}})$$

where:

$$d(\mathbf{Q},F_{\mathbf{a}}) = \sum_{ij}(\mathbf{Q}_{ij} - F_{\mathbf{a}}(\frac{i}{b^p},\frac{j}{b^p}))^2$$

3.4. Solving the non-linear fitting problem

Our resolution method is based on the LEVENBERG-MARQUARDT algorithm [13]. This algorithm is a numerical resolution of the following fitting problem:

$$\mathbf{a}_{opt} = \underset{\mathbf{a}}{\operatorname{argmin}} \sum_{i=0}^{M} (v_i - f(\mathbf{a}, u_i))^2$$

where vectors $\mathbf{v}$ and $\mathbf{u}$ are the fitting data and f is the fitting model.

In order to resolve our approximation problem using this algorithm, we have to consider the following data:

$$\mathbf{v} = (v_0, \ldots, v_M) = (0, \ldots, 0)$$

$$\mathbf{u} = (u_0, \ldots, u_M) = (0, \ldots, M)$$

where $M = (b^p + 1)^2$.

Then, the fitting model is:

$$f(\mathbf{a}, k) = \mathbf{Q}_{i_k j_k} - F_{\mathbf{a}}(\frac{i_k}{b^p}, \frac{j_k}{b^p})$$

where:

$$\begin{aligned} i_k &= k \mod b^p \\ j_k &= k / b^p \end{aligned}$$

$\forall k = 0, \ldots, (b^p + 1)^2$.

The LEVENBERG-MARQUARDT method combines two types of approximation for minimizing the square distance. The first consists in a quadratic approximation. When this fails, the method tries a simple linear approximation. These approximations are computed with the provided partial derivatives of the fitting model. In our case, these partial derivatives are numerically computed by a perturbation vector [7]:

$$\delta\mathbf{a}_i = (\underbrace{0, \ldots, 0, \varepsilon}_{i}, 0, \ldots, 0)$$

Then, the computation of partial derivatives is approximated by:

$$\frac{\partial f}{\partial a_i}(\mathbf{a}, u) \simeq \frac{f(\mathbf{a} + \delta\mathbf{a}_i, u) - f(\mathbf{a}, u)}{\varepsilon}$$

4. Results

The approximation method has been tested on three surfaces:

- A smooth surface: elliptic paraboloid shape
- A synthetic surface generated with our model
- A natural surface

4.1. Approximation of a smooth shape

Fig. 6 shows the original elliptic paraboloid shape and the approximated surface reconstructed with our method. The equation of this shape is:

$$z = z_0 - (\frac{x - x_0}{\sigma_x})^2 - (\frac{y - y_0}{\sigma_y})^2$$

Note that the original and the approximated shapes are very similar. Our model allows to reconstruct smooth surfaces[5], and not only rough surfaces.

4.2. Approximation of a synthetic surface generated with our model

For this example, we have fixed a shape model (IFS and control points). This model has been used to generate a $3D$ shape. Then, the challenge is to recover exactly the original model only using the synthetic shape. Fig. 7 shows the original synthetic surface that has been generated with our model and the approximated surface reconstructed with our method. The model is successfully reconstituted. This validates our method. The original surface and the approximated one are very similar.

4.3. Approximation of a natural surface

Fig. 8 shows our first experiments on a natural surface. The original surface has been extracted from a geological database (found at the United States Geological Survey Home page `http://www.usgs.org`). The general aspect of the approximated surface is similar to the original one. Fig. 9 shows the convergence of the approximation process. The first column contains the 3D view of the surface at a given step. The second column gives the error surface between the original surface and the approximated one. The last column shows information about the iteration step, the global distance between the surfaces, and the computing time. We can see that few iterations is needed to obtain convergence (only seven in our example) and a total of 132 seconds computing time on Pentium III 450 MHz 64 Mo hardware.

5. Conclusion

We presented a new approach based on a fractal model named projected IFS model for approximating both smooth and rough surfaces. This is a hybrid model: IFS operators yield self-similar, complex surfaces; the control points project these surfaces on the original one in order to minimize the approximation error. The projected IFS model is in fact a parametric description which has the advantage of compactly describing the surface shape, making it useful for surface compression and reconstruction (grey-level image, range data).

References

1. R M Bolle and B C Vemuri, "On Three-Dimensional Surface Reconstruction Methods," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, no. 1, pp. 1–13, January 1991. 1

2. B Girod, G Greiner, and H Niemann, Eds., *Principles of 3D Image Analysis and Synthesis*, Kluwer Academic Publishers, 2000. 1

3. D Terzopoulos and D Metaxas, "Dynamic 3D models with local and global deformations: deformable superquadrics," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, no. 7, pp. 703–714, July 1991. 1

4. A Gupta and R Bajcsy, "Volumetric segmentation of range images of 3D objects using superquadrics models," *CVGIP: Image understanding*, vol. 58, no. 3, pp. 302–326, November 1993. 1

5. C. E. Zair and E. Tosan, "Fractal modeling using free form techniques," *Computer Graphics Forum*, vol. 15, no. 3, pp. 269–278, August 1996, EUROGRAPHICS'96 Conference issue. 1, 2, 4

6. C E Zair and E Tosan, "Unified IFS-based Model to Generate Smooth or Fractal Forms," in *Surface Fitting and Multiresolution Methods*, A. Le Méhauté, C. Rabut, and L. L. Schumaker, Eds., pp. 335–344. Vanderbilt University Press, Nashville, TN, 1997. 1, 2

7. E Guérin, E Tosan, and A Baskurt, "Fractal coding of shapes based on a projected IFS model," in *ICIP 2000*, Sept. 2000, vol. II, pp. 203–206. 1, 3, 4

8. E Guérin, E Tosan, and A Baskurt, "A fractal approximation of curves," *Fractals*, vol. 9, no. 1, March 2001, To appear. 1, 3

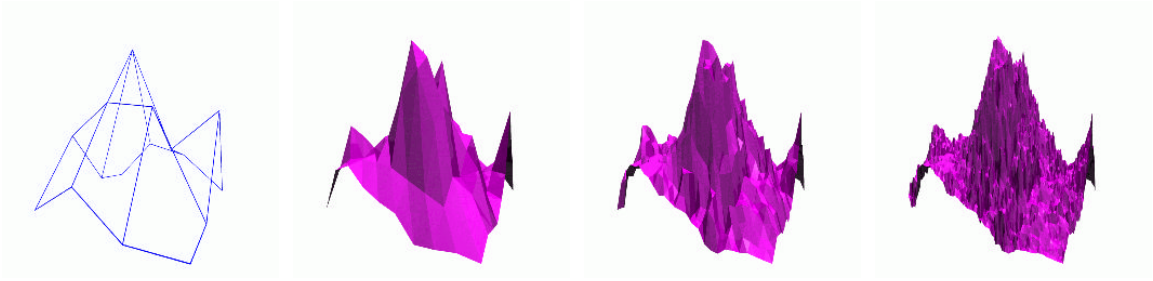
9. M. Barnsley, *Fractals everywhere*, Academic Press, 1988. 1, 2

10. A E Jacquin, "Image coding based on a fractal theory of iterated contractive image transformations," *IEEE Trans. on Image Processing*, vol. 1, pp. 18–30, Jan. 1992. 1, 2

11. E Tosan, "Surfaces fractales définies par leurs bords," in *Journées "Courbes, surfaces et algorithmes", Grenoble*, L. Briard, N. Szafran, and B.Lacolle, Eds., 15-17 Septembre 1999. 2

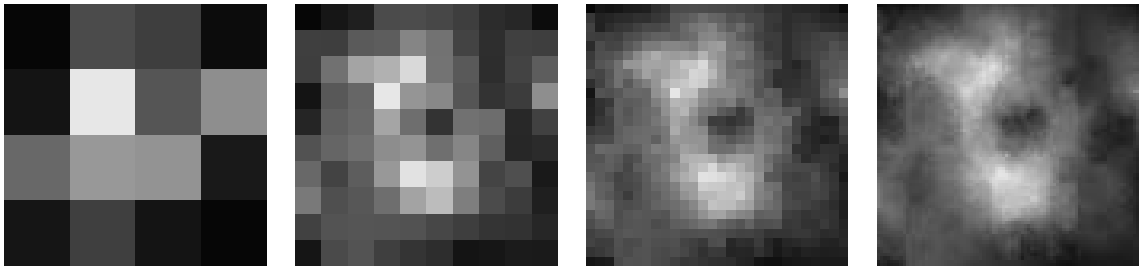
12. C A Micchelli and H Prautzsch, "Computing surfaces invariant under subdivision," *Computer Aided Geometric Design*, , no. 4, pp. 321–328, 1987. 2, 3

13. W H Press, B P Flannery, and S A Teukolsky andW T Vetterling, *Numerical Recipes in C : The Art of Scientific Computing*, Cambridge University Press, 1993. 4



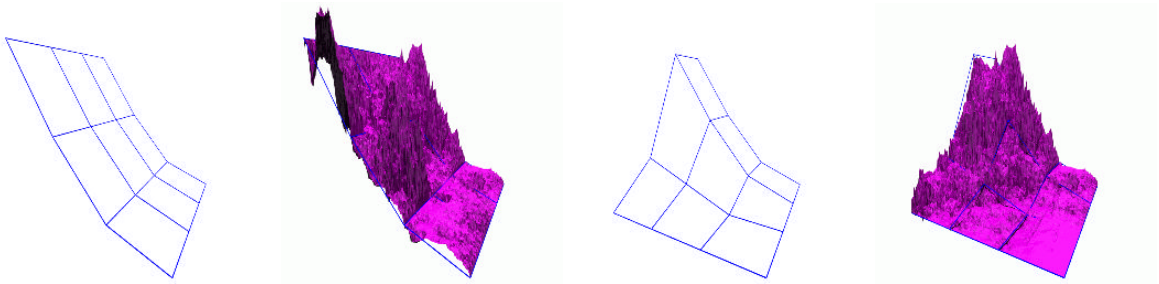
Step 0 Step 1 Step 2 Step 3

Figure 2: *Three first iterations of the construction process beginning with the initial* 4×4 *control grid (Step* 0*).*



Step 0 Step 1 Step 2 Step 3

Figure 3: *Three first iterations of the construction process beginning with the initial* 4×4 *image (image point of view).*



With control grid 1 With control grid 2

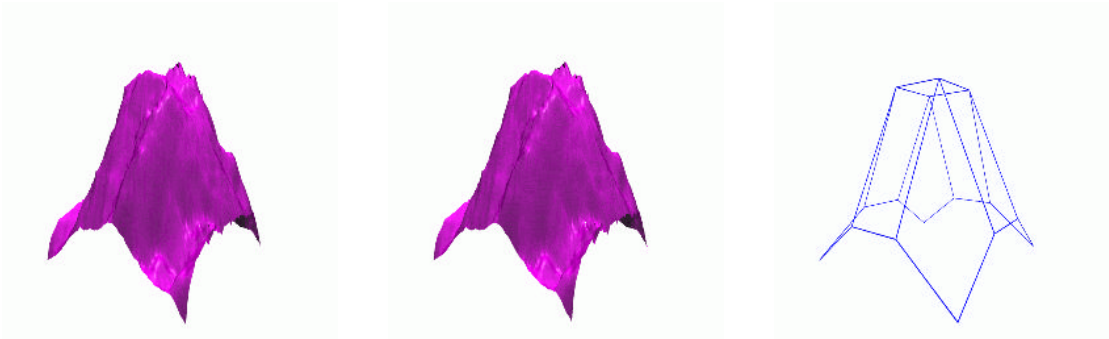
Figure 4: *Deformation of surfaces using the control grid.*

With control grid 1 With control grid 2

Figure 5: *Deformation of surfaces using the control grid (image point of view).*

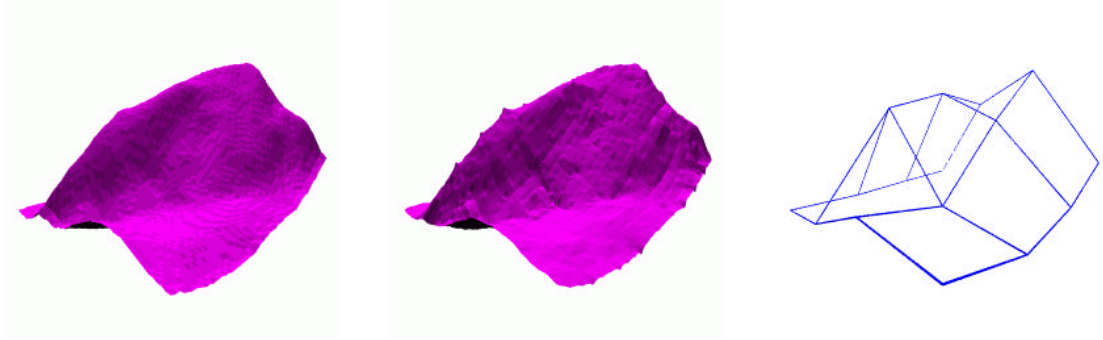
Original smooth shape | Approximated surface | Control grid

Figure 6: *Result on a smooth shape.*



Original surface | Approximated surface | Control grid

Figure 7: *Result on a synthetic surface generated with our model.*



Original geological surface | Approximated surface | Control grid

Figure 8: *Result on a geological surface.*

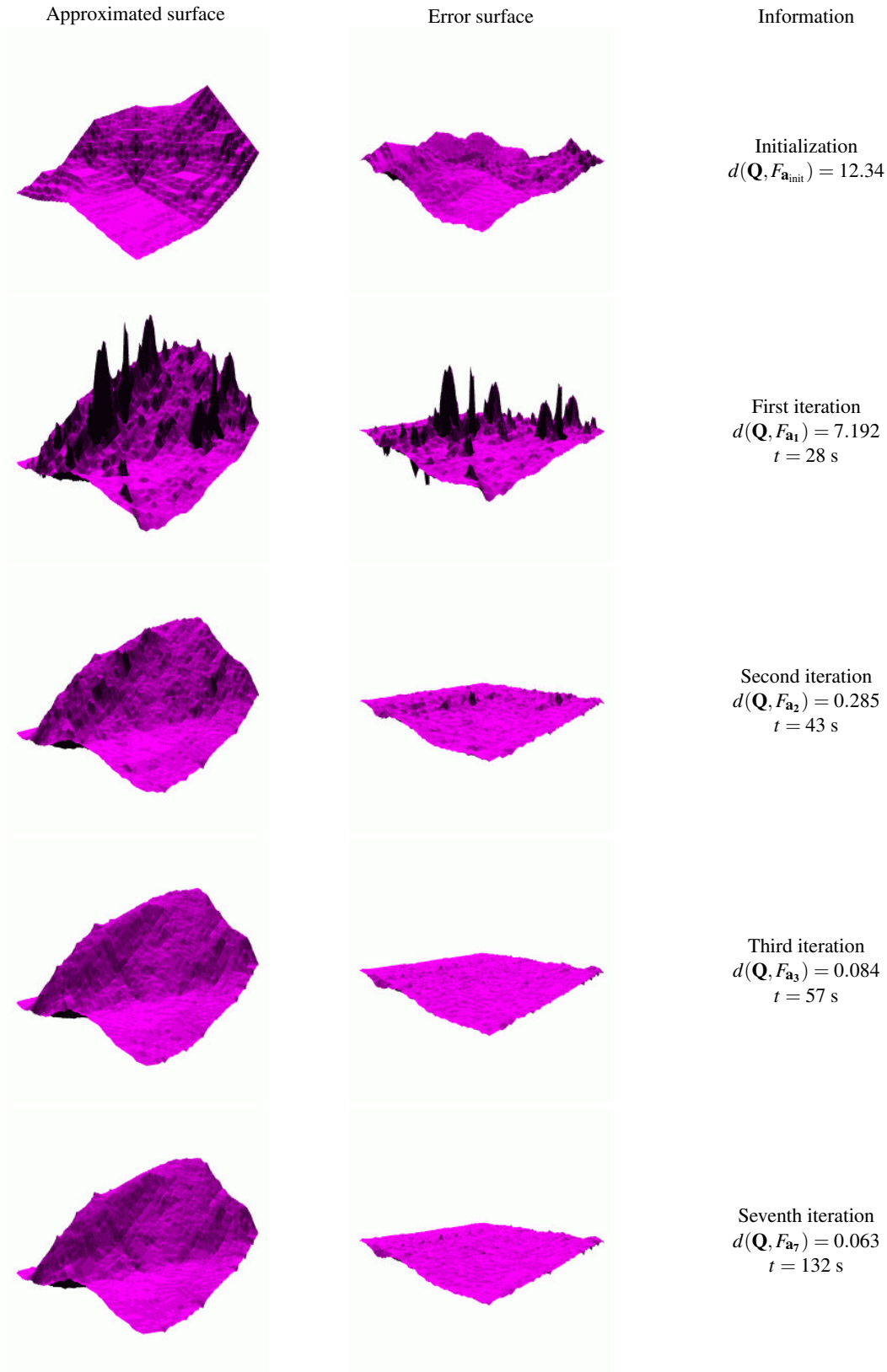


Figure 9: *Convergence of the approximation.*