# Locally Constrained Synthetic LoDs Generation for Natural Terrain Meshes

M. Perez [a], M. Fernandez [a], P. Morillo [a] and I. Coma. [a]

[a]Robotics Institute,University of Valencia,
P.O. Box 2085, 46071 Valencia, Spain

Terrain representation is a basic topic in the field of interactive graphics. The amount of data required for a good quality of the terrain offers an important challenge to developers of such systems. For users of these applications, the accuracy of geographical data is generally less important than its natural visual appearance. This makes it possible to maintain a limited geographical database for the system and to extend it generating synthetic data. The evaluation of the intrinsic properties of the terrain (i.e. fractal dimension, roughness, etc.) may be used as the basis for generating extra data accomplishing the same patterns discovered in the actual information. However, it is also interesting to point out that in most natural landscapes, it is usual to have human or natural changes in the basic properties of some areas, i.e. a road or a river. This fact can make it more difficult for synthetic data generation to be free of visual artifacts within these areas. In this paper we combine fractal and wavelet theories to provide extra data which keeps the natural properties of actual information available. New levels of detail for the terrain are obtained by applying an inverse Wavelet Transform to a set of values randomly generated, thus maintaining the coherence of statistical properties with the original geographical data. Combined with this approach, the use of energy reduction masks has been added in order to avoid undesired visual artifacts in those special areas for which the general terrain properties are no longer valid.

*Keywords:* Fractal, Wavelets, Terrain mesh, Synthetic extra resolution generation, Locally constrained extra resolution.

## 1. Introduction.

Terrain representation is a basic topic in the field of training simulators, in both military and civil applications. It is obvious that, in these representations, the amount of geographical data provided to the subject is clearly related to the feeling of visual immersion achieved by the system. However, it is not so obvious that the geographical data accuracy is less important than their natural visual appearance. Usually, terrain representation is based on a digital elevation model (DEM) plus a set of textures mapped on the mesh. The viewer integrates this geographic information without realizing the real source of the data presented except for the natural appearance of the final representation.

Another important topic in simulation and interactive graphics is freedom of movement within the synthetic environment, which should be as similar as possible to the real world. Additionally, it is expected to have an homogeneous visual quality from every possible point of view. This introduces an important drawback in most of the current terrain representation models, which have a limited resolution for the viewer. For instance, a flight simulator offers a good quality terrain representation appearance when observed from relative high elevations. However, for low height fly missions, the visual quality of the representation suffers from an important degradation, because the original terrain model is not accurate enough to increase the details of the representation.

A possible solution to this problem could be to resize the definition of the geographical data base by adding extra data for special purposes. Nevertheless, there will always be some limitations which come mainly from budget restrictions, availability (it is not always possible to have access to the data with the required accuracy), resources consumption (storage space, computing,

etc.). An approach to solve the resizing problem could be adding extra resolution to real terrain meshes by generating new synthetic data naturally. In order to maintain the natural appearance and the fidelity to the original terrain, these extra data should statistically follow the properties extracted from the original data. To accomplish this goal we propose a combination of fractal (in particular $1/f$ processes) and wavelet theory.

The previous approach has however an important drawback: most real situations include parts of the terrain with areas naturally or artificially modified from the general fractal pattern present in the terrain (rivers, lakes, highways, villages, etc). If those elements are not taken into account in generating new data, visual artifacts may appear (i.e. an island that does not really exist in the middle of a lake, a part of a road covered by the terrain -see figure 12 top-, unnatural river lines -see figure 13 top-, etc.). So, it is necessary to modify the previous idea in order to avoid these undesired visual artifacts.

The paper is organized as follows: in the next section of this paper a short overview of the fractal and wavelet theoretical bases used in this work is presented. In the third section the general approach that does not include any specific consideration for the special areas above indicated is introduced. In the fourth section, the basic approach is modified in order to avoid visual artifacts in the special areas. Finally, the paper closes with the conclusions and some future work.

## 2. Theoretical Background

### 2.1. Fractal and Multifactal Functions

Stochastic fractals, in particular the family of $1/f$ fractals, have been successfully used to model a wide variety of natural phenomena and even the sociological aspects of human behavior. Well known examples are[15].: natural landscapes, distribution of flow of a river's turbulent flow, the evolution of the stocks in the markets, etc.

An important characteristic of the $1/f$ processes is that its spectral density presents an exponential behavior. The spectral density follows

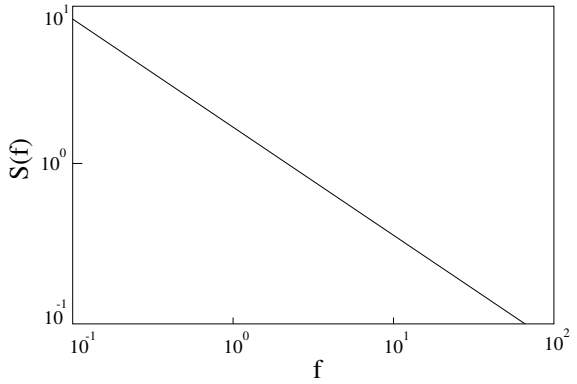


Figure 1. Representation of the $1/f^{\gamma}$ process spectral density .

the equation:

$$S(f) \sim \frac{1}{f^{\gamma}}$$

where $f$ is the frequency and $\gamma$ is a constant.

This means that a log-log representation of this density with respect to the frequency is a straight line with a slope $-\gamma$ (figure 1).

*Fractional Brownian motion (fBm)* is probably the best known mathematical characterization of the $1/f$ processes [7]. This theory has been frequently studied because of its simplicity and the wide range of natural phenomena that is able to model.

The most interesting properties of fBm processes are:

1. Statistically self-similarity, independently from scale.

2. Nonstationary behavior, with high degree of correlation. The Hurst parameter $H$ measures autocorrelation and is directly related to the $\gamma$ index in the $1/f$ processes through the equation:

   $$\gamma = 2H + 1$$

3. Stationary variations of a function $X(t)$ are

characterized by the equation:

$$\mathrm{Var}(X) \propto \tau^{2H}$$

being $\tau = |X(t+\tau) - X(t)|$.

Following [7-11], fractal objects can be represented by their fractal dimension. This parameter has an integer value for non-fractal objects (value 1 for curves, value 2 surfaces, etc.), and has a non-integer value that is bigger than its topological dimension for fractal objects. Fractal dimension can be used as a roughness indicator. For instance, fractals with fractal dimension close to two represent surfaces with a low roughness appearance while objects with fractal dimension values close to three correspond to extremely rough appearance. It is also possible to use this qualitative relationship in the opposite way. We can have a clue about the fractal dimension of an object based on its roughness appearance.

FBm fractal functions are characterized by having only one fractal dimension with an homogeneous distribution over the whole domain. This type of fractal objects are usually known as monofractals.

There are other kinds of fractals which exhibit a variation of the fractal dimension along its domain; this family of fractals is usually referred to as *multifractals*. Multifractals were firstly introduced to model energy dissipation turbulence [8,5]. They have been proved to be adequate to model a wide range of non-homogeneous processes [11,10].One important consequence of the fractal dimension variability across the object is that it is not longer possible to represent this object by a unique fractal dimension value because it changes in an erratic way along the object domain.

### 2.2. Wavelets on the Generation of fBm.

As previously pointed out, two of the most interesting properties of the fBm noise are its non stationary behavior, with stationary increments, and its self-similarity at different scales. The stationarity property requires a time-dependent analysis, while self-similarity requires some scale-dependent analysis. These two characteristics are presented in the wavelet transform [6,2], which makes this mathematical approach a powerful tool for analysis and synthesis of fBm's.

An orthonormal wavelet decomposition of a function $X(t)$ generates detail and coarse coefficients recursively by using the equations:

$$\begin{array}{l} c_{n,i} = \sum_k c_{n+1,k} h[-i+2k] \\ d_{n,i} = \sum_k c_{n+1,k} g[-i+2k] \end{array} \quad (1)$$

being $c_{n,i}$ and $d_{n,i}$ the coarse and the detail coefficients, respectively and $h[\cdot]$ and $g[\cdot]$, the low and high filter coefficients associated with the wavelet base.

The statistical behavior of these coefficients was previously analyzed by Faldrin [4], who draw the following conclusions that can be applied to this piece of research:

1. The sequence of the detail coefficients at any given level is self-similar and stationary. The non-stationary behavior of fBm is reflected only in the coarse coefficients.

2. The correlation amongst detail coefficients at any given level is only function of the distance between any two of them and decreases asymptotically.

3. The detail coefficients variance decreases following a power law with respect to the level:

   $$\mathrm{Var}\left[d_{n,i}\right] = V_\psi(H)\, 2^{-n\gamma}$$

   being $V_\psi(H)$ a function depending on $H$ and on the selected wavelet base. As a consequence of this behavior, the variance logarithm is a straight line with -$\gamma$ slope (figure 2):

   $$\log_2\left(\mathrm{Var}\left[d_{n,i}\right]\right) = -n\,\gamma + const$$

4. Even though fBm functions have a high correlation , specially those with $H$ parameters close to 1, the detail coefficients present a low correlation degree.

Among the previous conclusions drawn from Faldrin's works, the most appealing for our purposes is the fourth. As pointed out by Wornell in

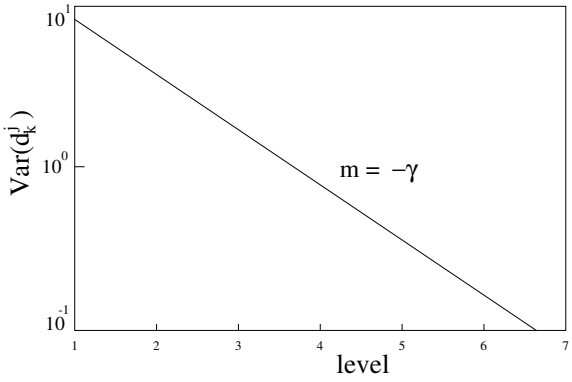


Figure 2. Influence of level in the variance of the detail coefficients.

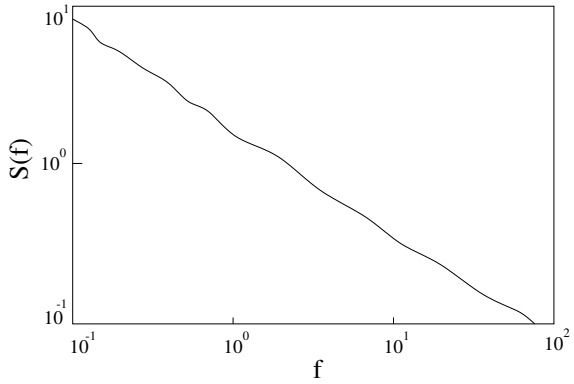


Figure 3. Spectral density of *nearly-fBm* functions.

[16], this property allows the correlation amongst the detail coefficients to be ignored in practical applications. Therefore, it is possible to consider these coefficients as a set of independent Gaussian variables with variance:

$$\sigma_n^2 = \mathrm{Var}\left[d_{n,k}\right] = \sigma^2\, 2^{-n\gamma} \tag{2}$$

The synthesized function obtained applying the Inverse Wavelet Transform to this set of Gaussian detail coefficients has an spectral density quite similar to fBm. This represents quite a simple procedure to synthesize nearly-fBm functions. The similarity between the results obtained through this method and those obtained by means of a pure fBm is directly related to the number of "vanish moments" of the wavelet base selected. In spite of this, it is not necessary to use basis with high number of vanish moments. In fact, the only important restriction is to select wavelet bases with a regular MRA [15].

### 2.3. Terrain Representation Based on fBm.

Natural landscapes are examples of self-similar fractal phenomena where the geometrical structure is invariant when examined at different scales. This qualitative characterization is the basis for most approaches to synthetic terrain generation [12,1].

Several years ago Mandelbrot made the first attempts at generating fractal landscapes [9]. Mandelbrot realized the analogy between the Brownian motion and mountain contours. From this impression, he figured out that a generalization of fBm to surfaces would result in a proper approximation to the description of natural mountain chains.

If we consider $V_H(x, y)$ a Brownian surface with a Holder parameter $H$ $(0 < H < 1)$, every section of this surface obtained by a vertical plane generates a fBm curve with $H$ parameter. Based on this property, if we move a distance $\Delta r = \Delta x^2 + \Delta y^2$ along the surface, the expected value of the function variation is:

$$\mathrm{Var}(V_H) \propto \Delta r^{2H}$$

The fractal dimension of this surface is bigger than the topological dimension of the corresponding non-fractal surface.

## 3. Approaches to Increasing the Resolution of a Terrain Mesh.

After the introduction of some basic concepts about fractal functions and wavelets, we explain our approaches based on principles given to increase the resolution of a natural terrain mesh by adding new levels of detail (LoD). The new

synthetic LoDs preserve the statistical behavior intrinsic to real data.

In the field of terrain representation two kinds of meshes can be used: regular and irregular meshes. Our approach is based on the use of the Wavelet Transform. This transform performs better when applied to a set of discrete regular distributed values, so our work is restricted to the use of regular terrain meshes.

### 3.1. Mesh Reconstruction From Wavelet Coefficients.

In order to understand our approach, let us review the way in which the meshes are generated from the set of wavelets coefficients. The Inverse Wavelet Transform (IWT) generates the original surface $f^N(x,y)$ using the wavelet coarse and detail coefficients by means of the following expression:

$$f^N(x,y) = \sum_{k=-\infty}^{N-1} \sum_{i,j} \sum_{t=1}^{3} d^t_{k,i,j}\, \psi^{2,t}_{k,i,j}(x,y) \qquad (3)$$

where $\left\{\psi^{2,1}, \psi^{2,2}, \psi^{2,3}\right\}$ are the 2D wavelet base functions (horizontal, vertical and diagonal) and $\left\{d^1, d^2, d^3\right\}$ are the coefficients associated to the base functions.

The reconstruction of the original function can also be carried out by using a linear combination of the 2D wavelet ($\psi^{2,t}$) and scale ($\varphi^2$) functions:

$$\begin{array}{c} f^N(x,y) = \sum_{i,j} c_{0,i,j}\, \varphi^2_{0,i,j}(x,y) + \\ \sum_{k=0}^{N-1} \sum_{i,j} \sum_{t=1}^{3} d^t_{k,i,j}\, \psi^{2,t}_{k,i,j}(x,y) \\ = \sum_{i,j} c_{N-1,i,j}\, \varphi^2_{N-1,i,j}(x,y) + \\ \sum_{i,j} \sum_{t=1}^{3} d^t_{N-1,i,j}\, \psi^{2,t}_{N-1,i,j}(x,y) \end{array} \qquad (4)$$

In real time terrain representation, triangle meshes are usually used. It is possible to consider these meshes as linear discrete surfaces, described by a set of vertices which define a set of adjacent triangles. A particular kind of triangular meshes are the ones where the vertices are distributed over a regular grid.

When the Wavelet Transform (WT) is applied to such regular meshes, what is really computed is the values of the function $f^N$ (from equation 5) in the positions of the mesh vertices. This means the coordinates $(x,y)$ of $v_{i,j}$ vertex (where index i, j are the relative position of each vertex in the mesh). If we suppose a dyadic decomposition of the original function, the function $f^N$ can be associated to a square mesh of $2^N \times 2^N$ vertices.

If we start from the previous mesh representation and we add the set of wavelet coefficient belonging to the next level to the sum in equation 3 , the resolution of the associated mesh is increased up to $2^{N+1} \times 2^{N+1}$ vertices. The function including the new coefficients is:

$$\begin{array}{c} f^{N+1}(x,y) = \sum_{i,j} c_{0,i,j}\, \varphi^2_{0,i,j}(x,y) + \\ \sum_{k=0}^{N} \sum_{i,j} \sum_{t=1}^{3} d^t_{k,i,j}\, \psi^{2,t}_{k,i,j}(x,y) \\ = \sum_{i,j} c_{N,i,j}\, \varphi^2_{N,i,j}(x,y) + \\ \sum_{i,j} \sum_{t=1}^{3} d^t_{N,i,j}\, \psi^{2,t}_{N,i,j}(x,y) \end{array} \qquad (5)$$

### 3.2. Global Scope Approach (GSA).

GSA is based on the generation of nearly-fBm functions using the results of the statistical behavior analysis of the detail coefficients in pure fBm [16]. As previously indicated, the fBm has self-similarity at every scale. If we assume that terrain meshes are fBm, or at least they are close enough to it, the new LoDs generated using this technique preserve the statistical properties of the original natural mesh. The assumption implies that the variance in detail coefficients follows a decreasing power-law (equation 2).

Taking into account the bidimensional nature of meshes for WT terrain representation , three different types of coefficients will be generated: horizontal ($d^1_{n,i,j}$), vertical ($d^2_{n,i,j}$) and diagonal ($d^3_{n,i,j}$). The variances of these coefficients ($\sigma^1$, $\sigma^2$ and $\sigma^3$ -equation 6-) are independent between them. This implies that we should define three different $\gamma$ values: ($\gamma^1$,$\gamma^2$,$\gamma^3$).

#### 3.2.1. GSA Algorithm .

The algorithm consists of the following steps:

1. Evaluate the associated variances to the wavelet coefficients in each of the levels and for each of the bands (horizontal, vertical and diagonal). It was assumed that these values follow a Gaussian distribution centered at zero ($\mu = 0$). The dispersion $\sigma$ at level n is obtained by using the following

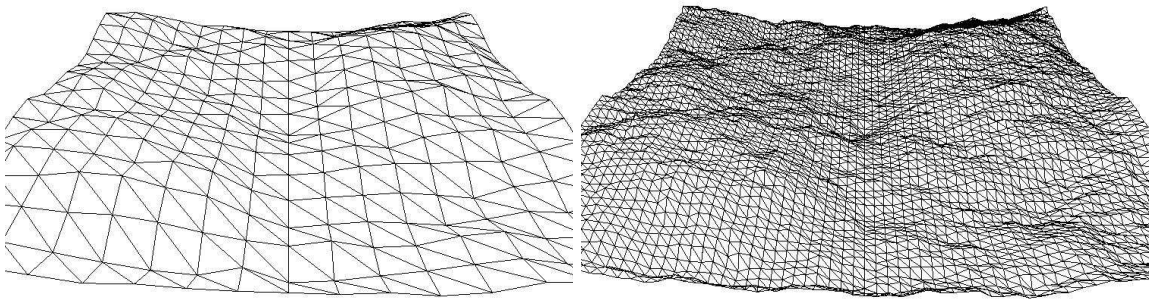

Figure 4. Application of GSA to an homogenous mesh. (left)original $(16 \times 16)$mesh; (right) final $(64 \times 64)$ mesh.

equations:

$$\left(\sigma_n^t\right)^2 = \mathrm{Var}\left[d_{n,i,j}^t\right] \tag{6}$$

2. Adjust the variances logarithm obtained in the previous step to a straight line:

$$\log_2\left(\mathrm{Var}\left[d_{n,i,j}^t\right]\right) = -n\gamma^t + a^t \tag{7}$$

The process has to be repeated for each of the three kinds of coefficients. This step generates three values: ($\gamma^1$,$\gamma^2$,$\gamma^3$). A WT dyadic decomposition is normally used, so the low levels do not have a number of coefficient values high enough to be considered statistically significant. As a consequence, we consider only variances from level 3 up, where we have at least 16 coefficients.

3. Generate randomly the values of the coefficients for the new levels. The generated values follow a Gaussian distribution centered at zero, with a dispersion obtained using equation 6. This guarantees the same statistical behavior as the one for the original levels.

4. Generate the terrain mesh new level by calculating the inverse wavelet transform of the coefficients obtained in step 3.

Figure 4 shows a visual example where the GSA approach has been applied to generate a finer resolution (from $16 \times 16$ to $64 \times 64$ vertices) of a natural terrain mesh. This approach offers good results when the fractal dimension of the original mesh is more or less homogenous across the whole surface (this is presented in figure 4). However, most practical cases do not follow the previous conditions. For instance, landscapes including rough mountains and smooth valleys have no homogeneous fractal dimension (figure 5).

### 3.3. Local Scope Approach (LSA).

To overcome these deficiencies, we have introduced the local scope approach that can be used in common DEMs, and which are better represented as multifractal objects.

Even though multifractals have been extensively studied, there are not too many real applications that use multifractal theory. A possible reason is the mathematical complexity involved in its use. Such remark is important because one of our goals is to develop an algorithm efficient enough to generate new LoDs dynamically. To accomplish both objectives -good characterization of the fractal nature of DEMs and good computational performance- we assume the following constraints:

1. The mesh is divided into regions. Each of them has a more or less homogeneous frac-

tal dimension.

2. The fractal dimension transition across neighbouring regions is smooth.

The first assumption does not impose important constraints because the fact that DEMs present a locally monofractal behavior has been well established in literature [1,9]. In fact, most analysis and synthesis algorithms related to terrain meshes are based on fBm. The second condition might be more difficult to accomplish in some cases, but its influence in the final result is not so critical since its partial infringement produces meshes that are still good enough.

**3.3.1. Local Fractal Dimension Estimation.**

In LSA, we have a different $\gamma$ parameter at each location. The $\gamma$ estimation is based on the use of the variance of the WT detail coefficients. However, we only consider the coefficients which have a spatial domain close to the location where we are estimating the $\gamma$ value.

Being $X(x, y)$ a function (accomplishing the previously stated conditions) to which we have previously applied the WT, we can conclude that around each point $\overrightarrow{r} = (x, y)$ there is a region that satisfies equation 6 with a single value of the $\gamma$ parameter.

As in the GSA, our meshes are discrete functions parametrized by two parameters. Then, WT produces three different types of detail coefficients: horizontal ($d^1_{n,i,j}$), vertical ($d^2_{n,i,j}$) and diagonal ($d^3_{n,i,j}$). The variances associated to each type of coefficients are independent of each other and can be derived from the equation:

$$\left(\sigma^t_n(\overrightarrow{r})\right)^2 = \mathrm{Var}\left[d^t_{n,i,j}\right] = A^t(\overrightarrow{r})\, 2^{-n\,\gamma^t(\overrightarrow{r})}$$

being $A^t(\overrightarrow{r})$ a constant through the different levels $n$ and $\gamma^t(\overrightarrow{r})$ the local $\gamma$ parameter at point $\overrightarrow{r}$.

The next step specifies the spatial regions with homogeneous fractal dimension. This determines the particular sets of coefficients to be used. Solving it in a accurate way is not trivial. It implies an additional computational cost that is incompatible with our temporal restrictions. To overcome this limitation we make an important simplification: actual regions are not determined. Instead, we suppose that the fractal dimension is more or less homogeneous in a square window centered at the current point, so only coefficients inside the window are used to estimate the $\gamma^t(\overrightarrow{r})$ parameters.

Once the window size has been selected, we always consider the same number of coefficients, no matter the level we are working on to obtain the local variance. The values ( $\gamma^1$, $\gamma^2$, $\gamma^3$ ) are calculated as the straight line slope represented by the equation:

$$\log_2(\mathrm{Var}\left[d^t_{n,i,j}\right]) = -n\gamma^t(\overrightarrow{r}) + a^t(\overrightarrow{r})$$

being $(a^1, a^2, a^3)$ the ordinates of the straight line.

Due to the fact that the spatial domain covered by detail coefficients increase exponentially when the level decreases, only the variance of the coefficients at higher levels has been considered.

**3.3.2. New LoDs Generation.**

Once the set of $\gamma^t(\overrightarrow{r})$ and $a^t(\overrightarrow{r})$ values has been evaluated, additional levels of detail are generated. To compute the vertices positions that conform the new LoD mesh at level $n$, we use the inverse wavelet transform applied to the synthetic coefficients generated randomly. The random generation is based on a Gaussian distribution centered at zero that has its dispersion expressed by the following equation:

$$\left(\sigma^t_n(\overrightarrow{r})\right)^2 = 2^{a^t(\overrightarrow{r})}2^{-n\gamma^t(\overrightarrow{r})} \tag{8}$$

The process can be repeated as many times as needed until we achieve the desired resolution in the final terrain mesh. This process can not be applied an unlimited number of times.

**3.3.3. LSA Algorithm.**

If we have $m$ initial levels generated using actual data, and if $k$ is the number of extra levels of detail (the final number of levels is $m+k$), the algorithm associated to LSA repeats the following steps for each point at level $m$:

1. Evaluate the local variance of the detail coefficients at levels $j < m$ (this process has to be repeated for the three types of coefficients).

2. Calculate the values of the slopes ( $\gamma^1(\overrightarrow{r})$, $\gamma^2(\overrightarrow{r})$, $\gamma^1(\overrightarrow{r})$ ) and the ordinate values ( $a^1(\overrightarrow{r})$, $a^2(\overrightarrow{r})$, $a^3(\overrightarrow{r})$ ), using equation 7.

3. Generate randomly, by using a Gaussian distribution centered at zero and with the dispersion obtained from equations 8, the new detail coefficients for levels from $m$ to $m + k - 1$ related to the current point $\overrightarrow{r}$.

Once the previous algorithm has been applied to every point at level $m$, the inverse wavelet transform is computed for the new added coefficients, obtaining the extra terrain levels of detail.

The quality of the results depends on three main parameters: the approach selected (local or global), the selected wavelet base and, in the case of a local scope approach, the window size.

Figure 5 shows an example of a heterogeneous mesh (a visual inspection of this mesh reveals different roughness areas along the surface), where the LSA algorithm has been applied. The results for different window sizes are shown in figure 6. In all cases, the same spline-linear base has been used. We can appreciate that the results for a $3 \times 3$ window size present a good visual quality. A more complete analysis of the results, and some considerations about performance and the influence of other wavelet bases, is described in [13,14].

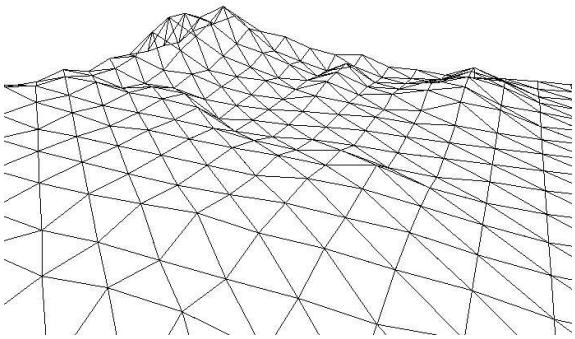

Figure 5. Example of heterogeneous mesh ($16 \times 16$) points.

## 4. Stochastic Locally Constrained LoD Generation

The GSA and LSA are based on the statistical properties present in the wavelet coefficients extracted from the real data to represent the terrain. The only thing that has been taken into account in both approaches is the wavelet coefficients that define the original terrain. Nevertheless, most real situations include parts of the terrain with areas naturally or artificially modified from the general fractal patterns present in the terrain (rivers, lakes, highways, villages, etc). If those elements are not taken into account in the generation of new LoDs, visual artifacts may appear (i.e. a nonexistent island in the middle of a lake, part of a road covered by the terrain -see figure 12 top- unnatural river lines -see figure 13 top-, etc.). So, it is necessary to modify the previous LoD's generation algorithms in order to include those considerations.

To overcome this problem, a local control over some areas of the new generated LoDs has to be considered. So, is introduced the concept of energy reduction masks. These masks absorb part of the energy associated to the wavelet coefficients generated when the GSA or LSA algorithms have been applied. This avoids the artifacts produced in the conflictive areas.

### 4.1. Locally Constrained Meshes Reconstruction.

If we start from the idea expressed at the end of point 3.1, and we assume that new coefficients values are zero, both surfaces extracted from the functions of equations 4 and 5 are identical: $f^{N+1}(x, y) = f^N(x, y)$. This does not mean that the higher resolution mesh surface, with $2^{N+1} \times 2^{N+1}$ vertices, extracted from the sampling of function $f^{N+1}(x, y)$, is identical to the mesh surface containing $2^N \times 2^N$vertices, extracted from the sampling of function $f^N(x, y)$. In fact, the only way in which both surfaces match is to use the biorthogonal spline-linear as wavelet base function for the representation.

Using this base function, the vertices of the new

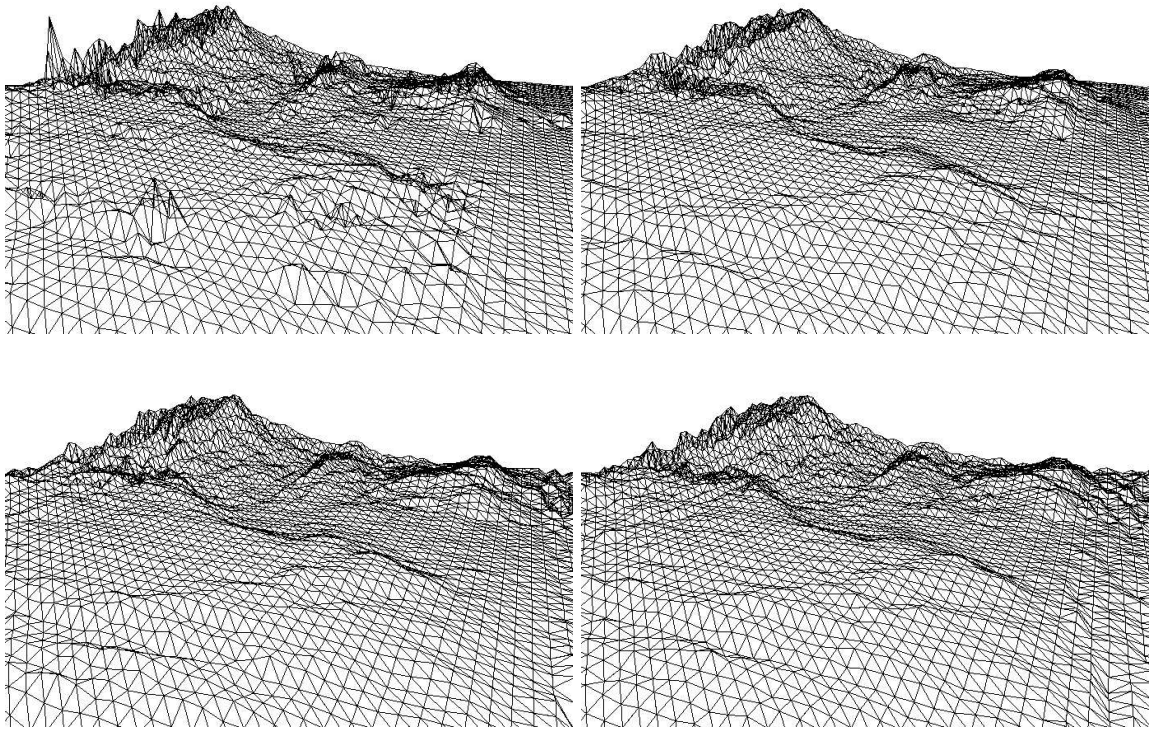

Figure 6. Results after having applied LSA to heterogeneous mesh with window size: $1 \times 1$ (left-top), $3 \times 3$ (right-top), $5 \times 5$ (left-bottom), and $7 \times 7$ (right-bottom).

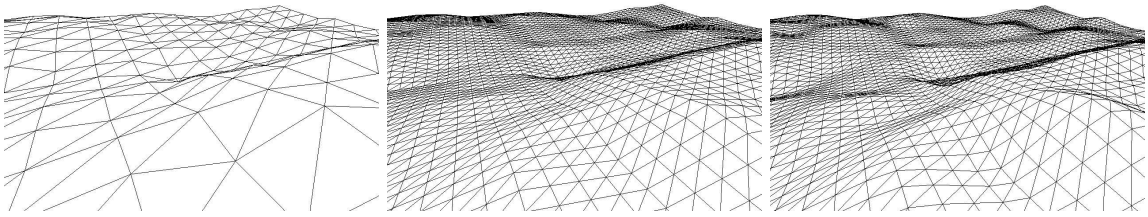

Figure 7. Left: Original Mesh. Middle: Extra resolution mesh generated with null values for extra detail coefficients and with linear-spline base. Right: Extra resolution mesh with null values for extra detail coefficients and with cubic-spline base.

mesh (extracted from the sampling of $f^{N+1}$, taking as zero the coefficients for the last level) are located over the surface defined by the previous level mesh (extracted from the sampling of $f^N$). The vertices of the new mesh which correspond to the same location of the vertices in the previous level mesh have the same altitude value, while the new elevation of the added vertices, which represent intermediate locations, correspond to the linear interpolation of its neighboring vertices (see figure 7, center).

It is important to point out that in the general case the vertices of the new mesh are not placed exactly at the same location as the one corresponding to the previous level mesh (see figure 7, right). This is accomplished only by spline-linear base, because it generates a linear surface similar to the one defined by a triangular mesh (see figure 7, center).

In order to avoid visual artifacts in the new surface at the areas modified by natural or artificial elements (rivers, roads, etc.), the approach is based on preserving the original surface shape for those parts of the terrain. To preserve the shape, it is just needed to eliminate the energy associated to the wavelet detail coefficients associated to the conflictive areas. This result can be accomplished with:

$$d^t_{N,i,j} = 0 \quad \text{if} \quad \left\langle \Omega\left(x,y\right), \psi^t_{N,i,j}\left(x,y\right)\right\rangle \neq 0$$

where $\Omega\left(x,y\right)$ is a function which is evaluated as 1 when $(x,y)$ is within the interest area and as 0 in the rest of the cases.

### 4.2. Mask Guided Reconstruction (MGR).

In order to eliminate the energy associated to the wavelet coefficients in specific areas, energy reduction masks are defined. These masks are built as gray-scale images. The gray level indicates the different areas on the terrain and the degree of energy absorption in each part of the original mesh (see figures 8 and 9). When the gray level of a pixel in the mask corresponds to white color, all the energy associated to the related wavelet coefficients is eliminated. In the case of a gray level corresponding to black color, all the energy is preserved. For intermediate gray levels, only the proportional part of the energy is reduced.

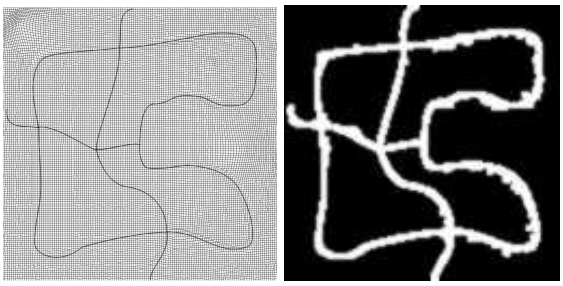

Figure 8. Left: Top view of the original road mesh. Right: Energy reduction mask defined to preserve the road area in this mesh.

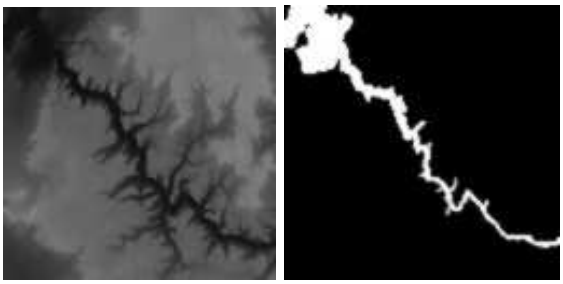

Figure 9. Left: Top view of second test mesh containing an applied texture with a river flow. The gray level indicates the elevation of each point. Right: Energy reduction mask to preserve the river area.

We can summarize the algorithm in the following steps:

- In the first step, the new wavelet coefficients are generated using the previously mentioned algorithms (GSA or LSA). These coefficients are the basis for the generation of extra LoD's for the terrain mesh.
- The second step uses the energy reduction masks. In this step, for each pixel $(k,l)$ in the mask, with a gray level $c_{k,l}$ (normalized

between zero and one), the related new coefficients are selected and modified using a combination between the gray level in the mask and the energy wavelet function at this location. The relative change in the energy for each coefficient is calculated using the following equation:

$$\left(\left(d^t_{n,i,j}\right)^2 - \left(df^t_{n,i,j}\right)^2\right) / \left(d^t_{n,i,j}\right)^2 = c_{k,l} \left\|\psi^t_{n,i,j}(x,y)\right\|_{(x,y)\in D(k,l)} \quad (9)$$

From such equation it is possible to obtain:

$$\left(df^t_{n,i,j}\right)^2 = \left(d^t_{n,i,j}\right)^2 \left(1 - c_{k,l} \left\|\psi^t_{n,i,j}(x,y)\right\|_{(x,y)\in D(k,l)}\right)$$

where $df^t_{n,i,j}$ is the new value for the wavelet coefficient $d^t_{n,i,j}$, and $\left\|\psi^t_{n,i,j}(x,y)\right\|_{(x,y)\in D(k,l)}$ is the normalized energy of the related wavelet function, within the spatial domain of the pixel $(k,l)$.

- The last step of the algorithm applies the IWT to the whole set of wavelet coefficient within the desired level, generating the new LoD terrain mesh.

Usually, the wavelet functions are not directly employed in these processes. Instead, it is easier to use a bank of associated filters [3], which is really the same principle employed in the fast wavelet transform (equation 1). In this case, the idea is not to evaluate the energy of the wavelet function, but to use the value of coefficients present in the associated filter at the points of interest.

A finite filter (spline-linear) is used to perform this representation. This keeps the algorithm cost at order $O(M)$, where $M$ is the number of pixel in the mask.

A detailed analysis of equation 9 reveals that the energy is not only reduced for the wavelet coefficients directly related to a certain pixel of the mask, but also to neighboring coefficients having wavelet function with energy inside the domain of the pixel. An important advantage of this fact is that smooth transitions of energy absorption are produced on the edges of conflictive areas. As a consequence, this yields a better visual appearance in the generated terrain meshes.

### 4.3. Results.

In order to have a qualitative evaluation of the approach proposed, some tests using two different terrain meshes have been carried out. The first mesh consist of a mountain road in terrain. The road is represented as a 3D object on top of the terrain that has been modified to simplify the road construction (see figure 10). The second test mesh contains a natural terrain modification produced by a river flow. In this case, the river is a part of the texture used to improve the visual quality of the terrain representation (see figure 11). In both cases, the initial meshes have a $128 \times 128$ resolution. Such resolution is increased up to $256 \times 256$ vertices during the testing process.

For the first test,where the terrain has a more or less homogeneous fractal dimension (except for the conflictive areas modified by the path of the road), the GSA algorithm has been used. In the second test, which includes the river, the terrain exhibit a non homogeneous roughness. So, in this case the LSA algorithm, that performs better in this situation (the window size is $3 \times 3$) has been used. In both tests, the wavelet base used is the spline-linear, which has demonstrated good performance in GSA and LSA algorithms [13,14]. Other important reason to use this base is the fact that the location of vertices for new levels will be preserved (or linearly interpolated) in the conflictive areas.

Figures 12 and 13 correspond to the road mesh test and the river mesh test, respectively.

## 5. Conclusions and Future Work.

We have introduced two algorithms to generate new detail levels of terrain meshes using the fractal properties extracted from real data representing the terrain. The first algorithm, GSA, is quite simple, but it is restricted to meshes with an homogeneous fractal pattern only (figure 4). To overcome this limitation, the second algorithm,LSA, introduces the local fractal estimation

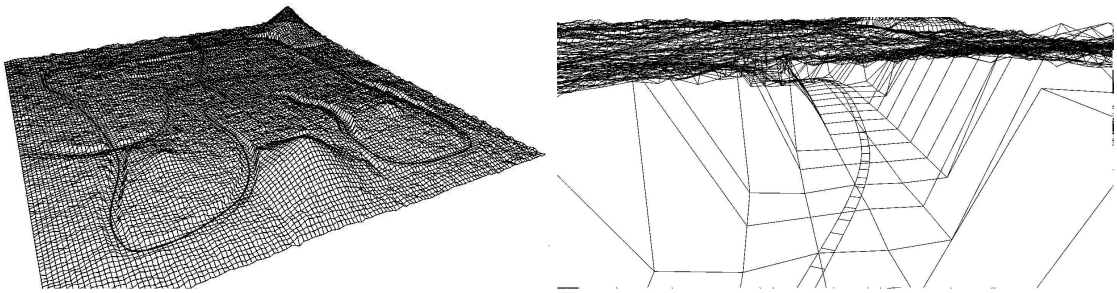

Figure 10. Left: Road terrain mesh ($128 \times 128$ vertices resolution). Right: A close-up of the road area.

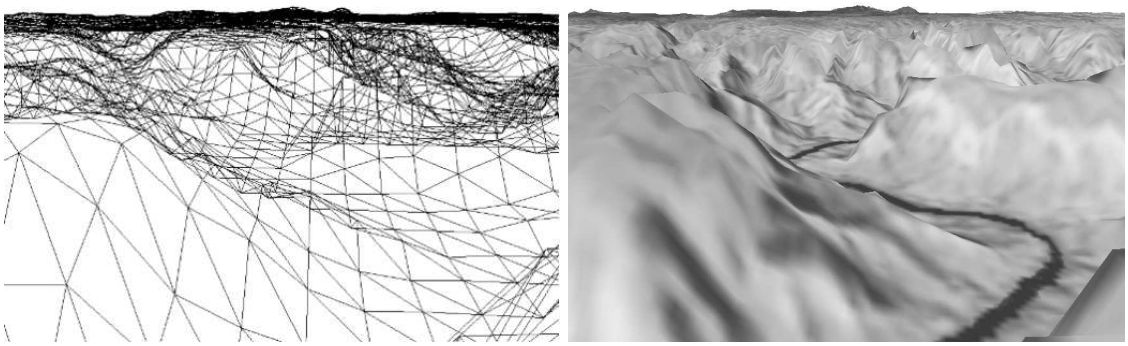

Figure 11. Left: A close-up of river terrain mesh ($128 \times 128$ vertices resolution). Right: Same view of the area with a texture mapped containing a river flow.

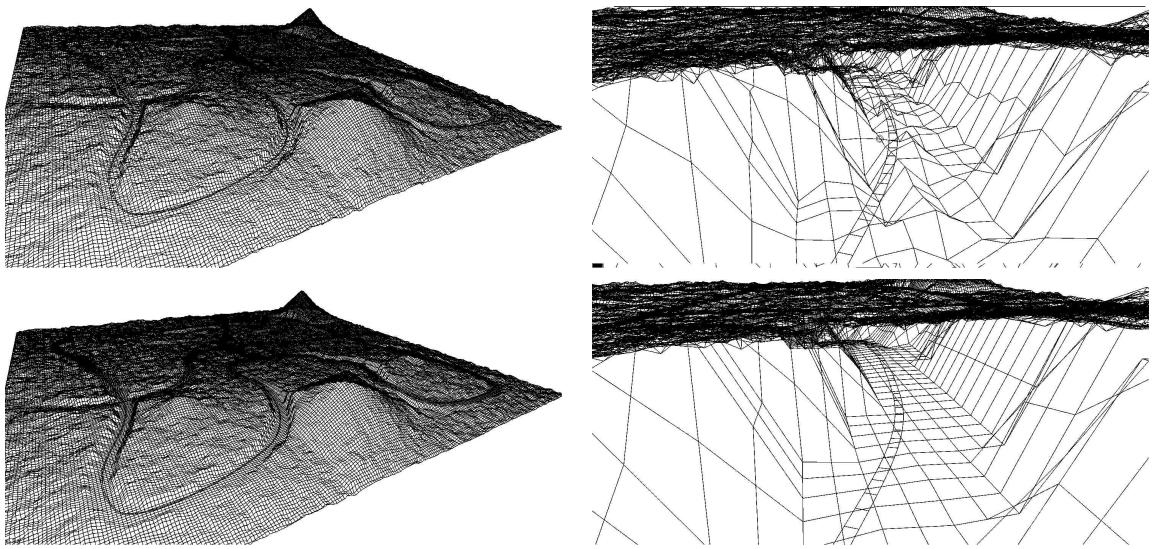

Figure 12. Left-top: Extra resolution mesh with $256 \times 256$ vertices obtained applying the GSA algorithm to the road test mesh without using MGR. Right-top: Details of the problems with extra vertices added that cover the original road element. Left-bottom: Extra resolution mesh obtained using GSA combined with MGR. Right-bottom: Details of the road where visual artifacts have disappeared.

which allows to apply synthetic LoDs generation to a wide range of terrain, as observed in figure 6). However, we have also indicated that general terrains include some special features that are not present in the fractal representation (figures 12 top and 13 top). To overcome the visual problem of artifacts we have introduced the energy reduction mask which complement GSA and LSA. As we can see in the Figures 12-bottom and 13-bottom, GSA and LSA including MGR are able to solve the visual problems.

We have performed some tests in order to estimate the extra spatial and computational cost produced by the introduction of MGR. Using this information and the theoretical analysis of the algorithms, we can conclude that the spatial and the computational extra cost of the combined algorithms (GSA+MGR or LSA+MGR) depend on the mask size. A mask with a similar resolution as the original mesh offers good results. In this case, we have an extra spatial cost of around 20%, while in the case of the computational cost we have an increase that is around 15%. These results are still fully compatible with the temporal and spatial constraints imposed by the previous version of GSA and LSA [13,14].

From the previous conclusions, we can summarize that the combined approach presented here offers a general solution that can be employed to improve the resolution of any variety of terrain meshes.

At the beginning of this paper, and also in one of the test examples used in this paper (the river case), we pointed out the importance of texture mapping to improve the appearance and detail in terrain representation. The fractal nature of terrain meshes is also present in real terrain textures. This, combined with the fact that regular meshes and texture images are both two-dimensional arrays of data, indicates that a possible extension of the study presented here could be the generation of finer resolution textures in realistic terrain representations.

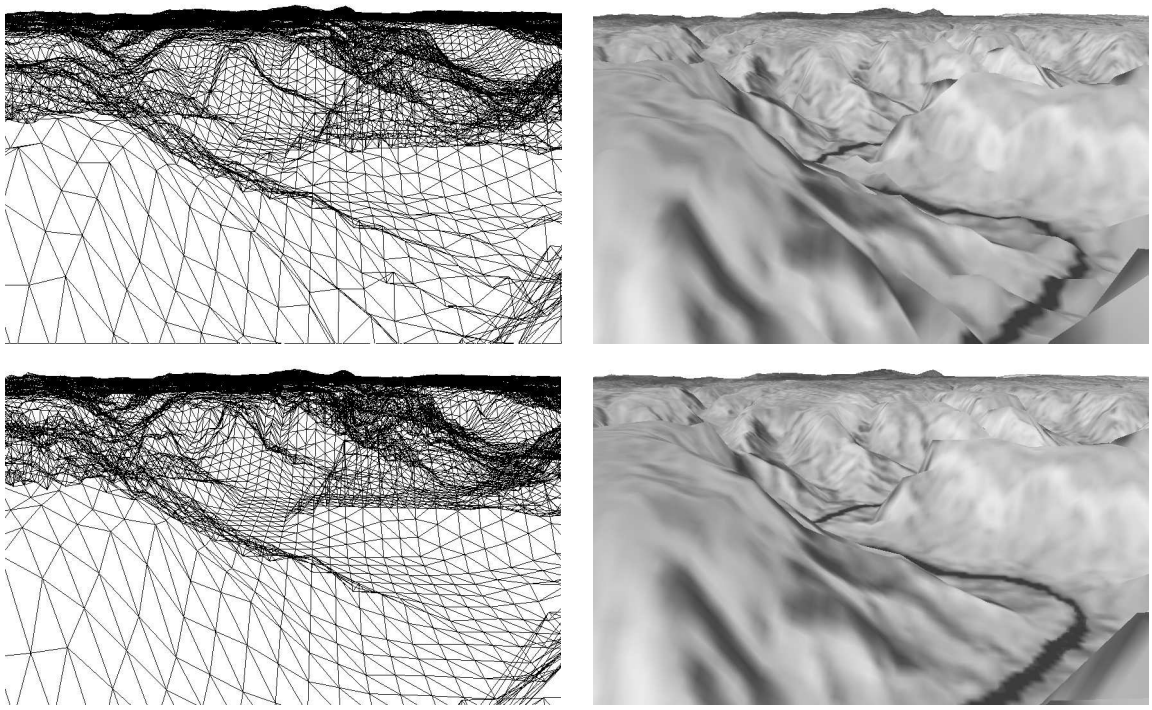

Figure 13. Left-top: Extra resolution mesh ($256 \times 256$) vertices obtained after having applyied LSA to the second test mesh. Right-top: Details of the problems that arise on the river area. Left-Bottom: Extra resolution mesh obtained with the combination of LSA and MGR. Right-bottom: Details of the improvement in the visual appearance of the river area.

## REFERENCES


1. M. F. Barnsley, R. L. Devaney, B. B. Mandelbrot, H. O. Peitgen, D. Saupe, and R. F. Voss. *The Science of Fractal Images*. Springer-Verlag, 1988. ISBN 0 387 96608 0.
2. I. Daubechies. Orthogonal bases of compactly supported wavelets. *Commun. Pure Appl. Math*, 41:909–996, 1988.
3. I. Daubechies. *Ten Lectures on Wavelets*. Society for Industrial and Applied Mathematics, Philadelphia, PA. USA, 1992.
4. Patrick Flandrin. Wavelet analysis and synthesis of fractional brownian motion. *IEEE Transactions on Information Theory*, 38(2):910–917, 1992.
5. U. Frisch and G. Parisi. Fully developed turbulence and intermittency. In *Int. Summer School on Turbulence and Predictability in Geophysical Fluid Dynamics and Climate Dynamics*, pages 84–88, 1985.
6. S. G. Mallat. A theory for multiresolution signal decomposition: the wavelet representation. *IEEE Trans. Patt. Anal. Machine Intell.*, 11:674–693, 1989.
7. B. B. Mandelbrot. Fractional brownian motion, fractional noises and applications. In *SIAM Review 10*, pages 422–437, 1968.
8. B. B. Mandelbrot. Intermittent turbulence in self similar cascades: Divergence of high moments and dimension of the carrier. *Fluid Mech.*, 62(3):331, 1974.
9. B. B. Mandelbrot. *The Fractal Geometry of Nature*. W. H. Freeman and Co., New York, 1982.
10. B. B. Mandelbrot. *Fractals and scaling in Finance*. Springer-Verlag, New York, 1997.
11. B. B. Mandelbrot and C. J. Evertsz. Multifractility of the armonic measure on fractal aggregates and extended self-similarity. *Physica*, A 177:386–393, 1991.
12. F. K. Musgrave, C. E. Bolb, and R. S. Mace. The synthesis and rendering of erosed fractal terrains. In *SIGGRAPH'89, Computer Graphics Proceedings*, pages 41–50, 1989.
13. M. Perez, M. Fernandez, and M. Lozano. Adding synthetic detail to natural terrain using a wavelet approach. *Lecture Notes in Computer Science*, 2330:22–31, 2002.
14. Mariano Perez. *Utilizacion de la transformada wavelet para la representacion de terrenos en entornos de simulacion*. PhD thesis, Universitat de Valencia. Departamento de Informatica, 2002.
15. Gregory W. Wornell. Wavelet-based representations for the 1/f family of fractal procesess. *Proceedings of the IEEE*, 81(10):1428–1450, 1993.
16. Gregory W. Wornell and Alan V. Oppenheim. Estimation of fractal signals from noisy measurements using wavelets. *IEEE Transactions on Signal Processing*, 40(3):785–800, 1992.



Mariano Perez has received his degree in physics from University of Valencia (Spain) and he holds his PhD in Computer Science from this University in 2002. Actually, he is an Assistant Professor at the Department of Computer Science (University of Valencia) and member of the ARTEC Graphics Group at the Robotics Institute. His research interest are computer graphics, virtual reality and real time terrain representation.

Marcos Fernandez is a Lecturer at the Department of Computer Science (University of Valencia - Spain) and Coordinator of the ARTEC Graphics Group at the Robotics Institute. He obtained his Msc. in Physics in 1992 and he holds his PhD. in Computer Science from the University of Valencia. He have been working in the field of real time computer graphics and virtual reality since 1992. Currently is interested in the area of highly immersive virtual environments

and he is involved in the development of a immersive visualization facility call VISIONARC. http://robotica.uv.es/grupos/artec/english.html

Pedro Morillo obtained a Msc. in Computer Science from the University of Valencia (Spain) in 1998. At present he is Assistant Professor of Computer Architecture at this university. His teaching and research interests centers on distributed virtual environments, collavorative virtual worlds and internetworking.

Inma Coma has received her degree in physics from University of Valencia (Spain). She has worked for 3 years as Assistant Professor and currently she is working as lecturer in the Computer Science Department of the University of Valencia. Her research interest are in computer graphics, virtual reality and driving simulation.