



**HAL**  
open science

## Generation of textures and geometric pseudo-urban models with the aid of IFS

Xavier Marsault

► **To cite this version:**

Xavier Marsault. Generation of textures and geometric pseudo-urban models with the aid of IFS. revues "Chaos and complexity letters" et "Chaos in Art and Architecture", 2005, pp.1-18. halshs-00260852

**HAL Id: halshs-00260852**

**<https://shs.hal.science/halshs-00260852v1>**

Submitted on 1 Apr 2008

**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.

# GENERATION OF TEXTURES AND GEOMETRIC PSEUDO-URBAN MODELS WITH THE AID OF IFS

*Xavier Marsault*

UMR CNRS MAP, "Modèles et simulations pour l'Architecture,  
l'urbanisme et le Paysage"

Laboratoire ARIA, Ecole d'Architecture de Lyon

3, rue Maurice Audin, 69512 Vaulx en Velin

Xavier.Marsault@lyon.archi.fr

<http://www.aria.archi.fr>

## ABSTRACT

Geometric and functional modelling of cities has become a growing field of interest, raised by the development and democratisation of computers being able to support high-demanding graphics in real time. Actually, more and more applications concentrate on creating virtual environments. ARIA has been working for two years, within the DEREVE project (DER, 2000), on pseudo-urban textures and geometric models generation, by means of fractal or parametric methods.

This paper explains our attempt to capture inner coherence of urban shapes and morphologies, by fractal analysis of  $2D\frac{1}{2}$  textures (top view + height) of real and synthetic city maps. The basic ideas lean on autosimilarity detection, fractal coding of regions, and processing with Iterated Function Systems (IFS). We introduce a genetic-like approach, allowing interpolation, alteration and fusion of different urban models, and leading to global or local synthesis of new shapes. Finally, a 3D reconstruction tool has been developed for converting textures to volumes in VRML, simplified enough for real time wanderings, and enhanced by some automatically generated garbage dump and decorated elements. Programs and graphic interface are developed with C++ and QT libraries.

**Keywords:** Fractal city, Urban pattern, IFS. Image, 2D½ and 3D model, Genetics, Fusion, Level of detail, Shape filtering, VRML

## 1. INTRODUCTION

### 1.1. Fractal Cities?

Usually, geometric models of town patterns or whole cities can be generated with the aid of spatial growth simulators, or temporal simulators based upon a scenario (ex: Sim City), or by means of static shapes (Parish, Muller, 2001). Many related works deal with fractality: some of them use cellular automata (Torrens, 2000), other ones use DLA (diffusion limited by aggregation) (Bailly, 1998) or organic models inspired by physical laws (Makse, 1996).

Indeed, some recent studies reveal the fractal nature of many urban structures at large scales and some architectural objects (Sala, 2002), (Batty, Longley, 1994), (Frankhauser, 1994). Focusing on the near scale of buildings and built patterns, we have shown that some urban shapes exhibit a local property of autosimilarity, while they lose it in a larger analysis. In this context, one way of research was to attempt to use IFS (that share this property) to analyse and generate new urban morphologies. Two cities belonging to the suburbs of Lyon (St Genis and Venissieux, fig 7a) and two synthetic maps (fig. 6) have been used during all developments and tests.

### 1.2. Iterated Function Systems (IFS)

IFS theory is totally based upon the “scale change invariance” property (SCE), and thus allow the generation of fractal objects with a set of contractive functions showing this property, called Iterated Function System (IFS). It has been studied by Hutchinson within the mathematical frame of autosimilarity (a mathematical object is said to be autosimilar if it can be split into smaller parts calculated from the whole by a “similar transformation”) (Hutchinson, 1981), and by Barnsley within the frame of fractal geometry (Barnsley, 1988), leading to image compression applications (Barnsley, 1992-1993), (Jacquin, 1992).

#### *Image Compression*

Since usually a given image is not a fractal object, it is unlikely to find a whole fractal generator of it. But there is an interesting application of IFS to image compression allowing fractal coding, where small regions are coded from

contractive SCE transformations (called *lifs*, for “local ifs”) of other larger regions. The most common algorithm was developed by Jacquin (Jacquin, 1992). Given a square uniform pavement of N range blocks  $r_i$  of size B and a pool of domain block  $d_i$  for matching research (Fig. 1), it tries to find a function  $i \rightarrow \alpha(i)$  and N *lifs*  $\omega_i$  such as  $r_i \rightarrow \hat{r}_i = \omega_i(d_{\alpha(i)})$ , so that  $\sum_i \|\omega_i(d_{\alpha(i)}) - r_i\|^2$  is minimum for each i (local collage). The famous “collage theorem” ensures that the decoded image is an approximation of the original image, and gives a maximum measure of the error. Each  $\omega_i$  is set with a transformation projecting a domain block  $d_i$  of size D at the place and scale of the range block  $r_i$  (decimation of pixels + isometry), followed by an affine transformation on grey levels of pixels :  $\hat{r}_i = iso_i(\sigma_i.r_i + \beta_i)$ . The isometric transformation  $iso_i$  is chosen among the 8 possible transformations of a square block (identity, -90, 90 or 180 degrees rotations, x, y, or diagonal axis symmetries).

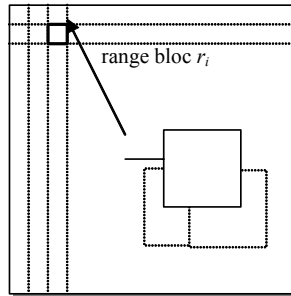


Figure 1: transformation from a domain block to a range block (lifs)

The image  $I$  is decoded by calculating the attractor of *lifs*  $w_i$  from any random image  $I_0$ .

$$\text{We note : } W(I) = \bigcup_{i=1}^N \hat{r}_i = \bigcup_{i=1}^N \omega_i(d_{\alpha(i)}) \text{ and } W^{k+1}(I) = W(W^k(I)).$$

$$\text{The attractor is then defined by : } A(I) = \lim_{k \rightarrow +\infty} W^k(I_0).$$

The more local collages are better, the more the result of the attractor is a good approximation of the original image (collage theorem). A little number of iterations is needed for the decoding process to converge. In order to ensure the

uniform convergence, we limit  $|\sigma_i| < 1$  for each block, even if it has been shown this limit can be exceeded (Hutgen, Hain, 1994).

### 1.3. Are Urban Patterns Autosimilar ?

This question appear to be essential, since one can rarely observe such a global property for objects in nature or in the real world, especially for urban structures. Nevertheless, the use of IFS for generating town-like patterns has been described by (Woloszyn P, 1998), who illustrates how the iteration of a simple substitution rule from an initial and basic pattern leads to an image that looks like an urban structure (Fig. 2). Fragmented patterns are recursively generated from only one transformation applied to all shapes ; they approximate the streets, the secondary networks, and the buildings, and are totally fractal. But this technique can't reproduce real city patterns, more irregular and more complex in their structure, and not totally autosimilar.

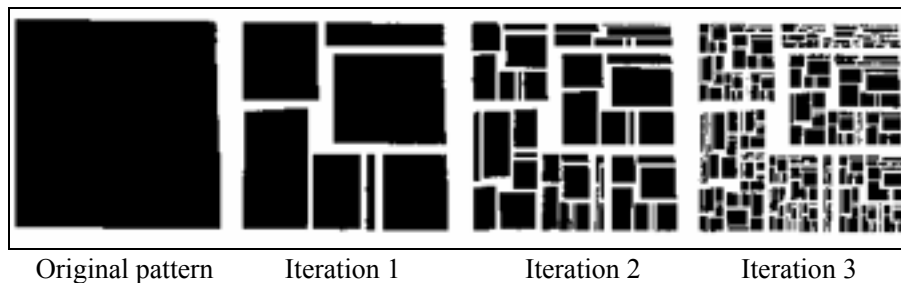


Figure 2: example of pseudo-urban fractals (Woloszyn, 1998)

On the other hand, a local property of autosimilarity can be more or less put in evidence with real urban patterns, and we can envision the application of local IFS to the generation of urban fabric. We've demonstrated this result by computing autosimilarity measures with Jacquin's IFS on 256 x 256 grey level images of urban patterns. In that purpose, we developed mathematical functions allowing some local or global measures of autosimilarity within an image, at the expense of computing time, because the algorithm still examines all square blocks configurations. The entire range and domain block pools  $\{B\}$  and  $\{D\}$  are parsed (not only those concerning Jacquin's pavements). For every pixel  $p \in B$ , all possible appariement configurations  $(B,D)$  are tested, avoiding the contraction limit  $\forall i, |\sigma_i| < 1$ , which speeds up the calculus. The appariement of  $D$  and  $R$

blocks is based upon the minimum of the local error measure defined by  $\mu(R, D) = \sum_{p \in B} (\omega_p(D) - R_p)^2$ , normalized between 0 and 1. The more  $\mu$  stretches towards 0, the better the appariement ; but it seems difficult to fix a threshold that decides there is autosimilarity or not. Therefore, we propose to define a local measure of this approximation, by searching the best couples (B,D) for appariement. Both cases with parameter B fixed or variable have been experimented.

When B is fixed, let's set up the D parameter, and an upper limit for D starting from B+1. In the case where D parameter is fluctuant, we keep a D block so that  $\{\mu(R, D) \text{ minimum}, D > B\}$ . Then, locally for each pixel p, we can define

an average autosimilarity measure :  $\tilde{\mu}(p) = \frac{1}{\text{card}\{R, R \supset p\}} \sum_{R \supset p} \min_{D > B} \mu(R, D)$ . But,

this average can potentially mask an existing D block for which the appariement is exact, or almost exact. So, we also calculate (Fig. 3) the minimal measure :

$\mu_{\min}(p) = \min_{R \supset p} \left( \min_{D > B} \mu(R, D) \right)$ . When B is fluctuant, we could locally consider the

higher value  $B_{\max}(p)$  of B for which the  $\mu_{\min}(p)$  measure is minimum, and propose another measure  $(1 - \mu_{\min}(p)) \cdot B_{\max}(p)$ , which grows both with B and the appariement quality. But such a task should require a tremendous computing time, even for a 256 x 256 image. One can also wander which information could be gathered from the study of the function  $D \rightarrow \mu(R, D)$ . For example, the decrease of this function could help characterizing a typical behaviour.

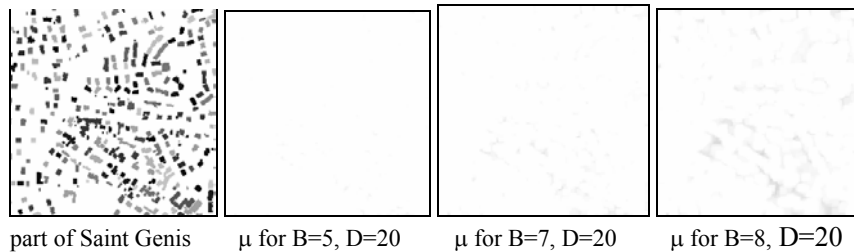


Figure 3.

Our goal being the generation of urban shapes and structures that look like real ones, we decide to lean on real city plans, and use the IFS as an analysis and synthesis tool. Because IFS operate on a continuous space of shapes, allowing interpolation, alteration and fusion, and integrate as a whole approach analysis

and generation of global or local new shapes, we expect them to produce good results.

## 2. FROM IFS TO URBAN TEXTURES AND GEOMETRIC MODELS

### 2.1. A Simplified Coding Method

Urban scale concerns the spatial distribution of buildings within a certain piece of landscape. It can be described with a restrictive approach by a set of more or less simplified volumes, especially for fast rendering. The image compression technique described in 1.2 allows an approximated coding of an image from local transformations of parts of itself. It can be used to encode urban pattern with IFS if we decide to convert geometrical 3D volumes to images. In this  $2dD^{1/2}$  approach, the grey levels represent the heights of buildings. Then, we use Jacquin's fractal compression technique for coding the ground shapes and heights of buildings which populate a city map. We get an autosimilar approximation of the map, whose accuracy depends on the nature and the choice of the initial pavement of the map, and on the number of local *lifs* given to approximate the local diversities of the shapes.

We have proposed some adaptations for urban pattern analysis: initial and static regular square pavement for the range blocks of size B, exhaustive research within the domain block pool (with varying size of D for each block B), pre-calculus of range blocks similarities, accelerated appariement by classification of range and domain blocks (uniform, outline), elimination of ground blocks and a « topological collage » for matching (see below).

#### *Towards a Spatial-Coded Model*

When several domain blocks D are candidates to the best appariement for a given range block B, the question of choice appears, whereas it is not significant in the frame of image compression. In that case, the algorithm should select the D block whose neighbourhood is the closest to the one of the B block. It expands successively the outline of each block by one pixel until it finds the minimum among all the proposed range blocks. This option that we call « *topological collage* » slows the processing time, but is the only one that can really take account of the topological links between B and D blocks. It has been successfully applied to our appariement algorithm used by the "asymmetric fusion" operator (see 2.3).

## 2.2. General Processing Scheme

Some pseudo-convex interpolation, mutation, fusion and filtering operators have been designed to generate new urban models leaning on existing ones (real or synthetic), or to add some modifications to them. For this purpose, we use the genetic analogy introduced by (Vences, Rudomin, 1997) (see 2.3).

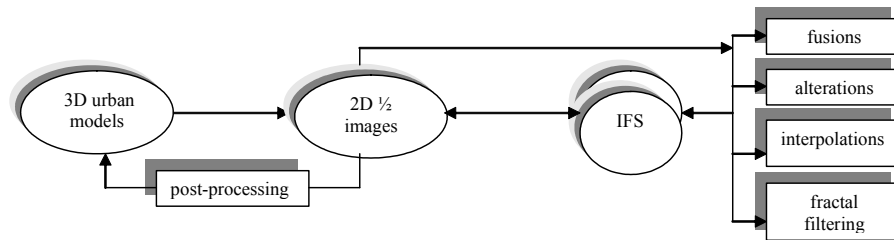


Figure 4 : general processing scheme

### *Pseudo-Convex Interpolation of IFS*

Given two distinct images encoded by  $IFS_1$  and  $IFS_2$ , our first idea was to define a  $\lambda$ -parametric convex IFS leaning on  $IFS_1$  and  $IFS_2$ . Indeed, if the iteration semi-group is convex, its attractor is  $\lambda$ -continuous (Gentil, 1992). As this is not the case with the semi-group composed by the 8 isometries of the square, we should rather speak of pseudo-convex interpolation. And the awaited results are disappointing, since we get in fact the same as basic image interpolation. Nevertheless, depending on other suitable choices for the type of pavement used, IFS interpolation could become possible.

## 2.3. A Genetic-Like Formalization

The genetic analogy proposed by Vences and Rudomin first in the frame of image compression, is very powerful for exploring new ways of creation, and let us envision applications to the generation and the alteration of urban geometric models. Assuming the notation  $IFS = (\omega_1, \omega_2, \dots, \omega_N)$ , where  $\omega_i$  are the *lifs*, we consider the IFS as a chromosome (genotype), the *lifs* as genes, and the attractor (image or 3D model) as a phenotype. This analogy can be justified in several ways. First, the information for decoding an image fragment is distributed among many *lifs*. Some *lifs* alterations can have consequences on numerous zones, or not. Moreover, the whole body of *lifs* represents a highly non-linear and complex system.



Following this scheme, we apply the general fusion mechanism (Renders, 1995) which consists in generating a large population of IFS models sharing the same genes (inherited from two parents), while the mutation allows the alteration of genes during the crossing process or the exchange of genes along the same chromosome. The following paragraphs describe some ways we used to implement the fusion process.

### ***Direct or Asymmetric Fusion***

We follow the genetics analogy, where the fusion process, even highly combinatorial, does not take any genes at random. Since *lifs* are coding zones whose content may be very different, the process of fusion must be guided by an appariement step between IFS parents. Indeed, without any control, the direct fusion *lifs* to *lifs* or their copy from one zone to another (a kind of mutation), give very bad results. So, we decided to keep the original distribution of *lifs* and to attempt to group them, before fusion occurs between both IFS.

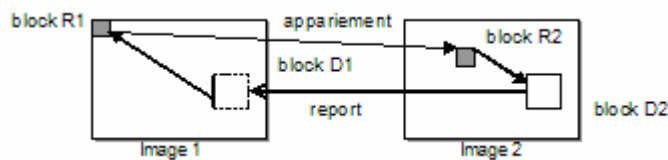
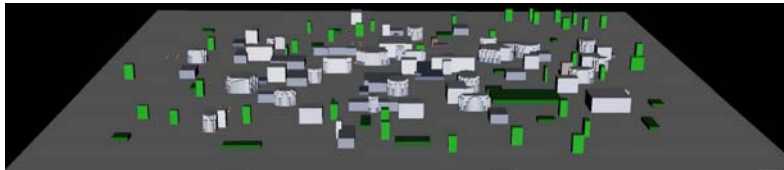
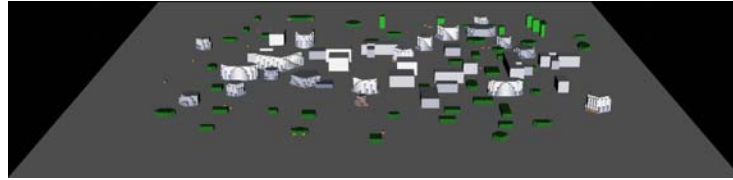


Figure 5 : principle of asymmetric fusion process

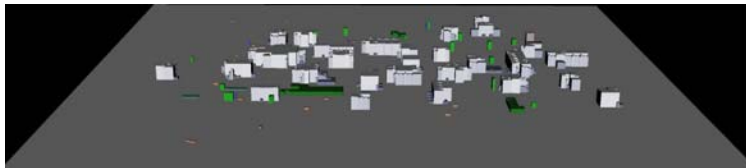
A first step we proposed to take this mechanism into account is to calculate appariements of range blocks between both images, based on the images content. The process is asymmetric, since we associate to each image a list of range blocks and their *lifs* counterparts in the other one (fig. 5). A range block  $R_1$  from image 1 is linked (by appariement) to a range block  $R_2$  of image 2, which is encoded by an *lifs* based on domain block  $D_2$ . The process of fusion consists in replacing *lifs*<sub>1</sub> by *lifs*<sub>2</sub>, and leads to two fusion images :  $1 \text{ mod } 2$  et  $2 \text{ mod } 1$ . One can observe on figure 6 the effect of asymmetric fusion : the generated distribution of shapes look like their two parents, modulated each one by the other.



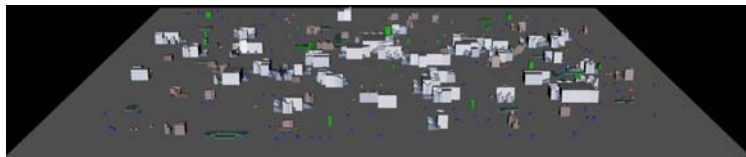
*synthetic town model A*



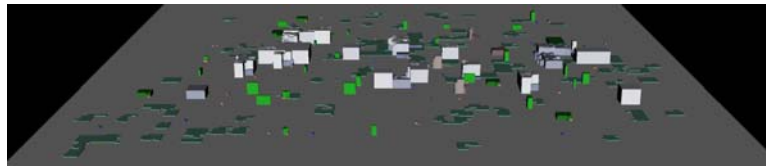
*synthetic town model B*



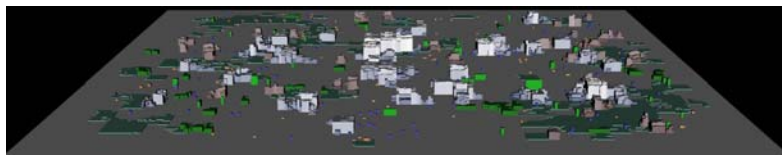
*asymmetric fusion AmodB for B=4 and D=8*



*asymmetric fusion BmodA for B=4 and D=8*



*asymmetric fusion AmodB for B=8 and D=16*



*asymmetric fusion BmodA for B=8 and D=16*

Figure 6 : 3D models of asymmetric fusion with synthetic towns A and B

***Pavement-Based Fusion***

Given a unique Jacquin square pavement for both IFS, and a size B for range blocks, we define square macro-blocks (or pavements) of size multiple of B, in order to group several connex range blocks. The pavement-based fusion consists

in crossing spatially grouped sequences of *lifs* (rather than isolated ones) between both IFS, in order to preserve topology. The process alternately keeps some *lifs* from the first IFS and the second one. Possible discontinuities only appears at the borders of the macro-blocks. Our technique lets the algorithm first inject some macro-blocks of important size, and finishes with smaller ones, like a town planner who first deals with higher scales of the city before looking at the content of the neighbourhoods. Moreover, the fact of varying the size of injected macro-blocks allows the modulation of the crossing scale. While varying the minimum and maximum limits of the macro-blocks size, we modify the model topology by authorizing more or less discontinuities.

The macro-block locations and the fill rates of each IFS are provided by the user or generated by a pseudo-random generator. The user also enters upper and lower limits for the macro-blocks size. The algorithms first fills up the entire image with one IFS. Then, it takes the other one, and will alternate until a break-test is verified. A margin is entered by the user to let the algorithm have a tolerance while matching the fill rate criterion. This margin progressively diminishes each time the filling IFS is changed, while the size of the macro-blocks is lowered of one-pixel. This is a heuristic allowing to create on the fly a new genotype from both parents' ones, guided by constraints depending on their phenotypes. For convenience, we added as fill rate criterion a fusion parameter  $\lambda$ , ranging from 0 to 1, allowing a sort of IFS interpolation between both models.

We define a non-intersection criterion allowing to label as "admissible" each macro-block whose outline does not intersect buildings more than a tolerance threshold  $S$ , given by the user. This criterion is computed on the grey level differences around the outline. Moreover, it also takes account for the previous crossing steps of the algorithm, leading to a better continuity in the phenotype shapes. Nevertheless, this precaution does not guarantee that all domain blocks will belong to preserved zones, but this is a first and serious limitation to this problem. The  $S$  parameter plays an essential role in the appearance of new and original shapes.

Pavement-controlled fusion gives some very good graphic, and provides new local or global shapes and distribution of shapes, whose details are the consequence of crossing models. The pavement choice is controlled by preservative criterions, and the number of map inputs is not limited for this process.

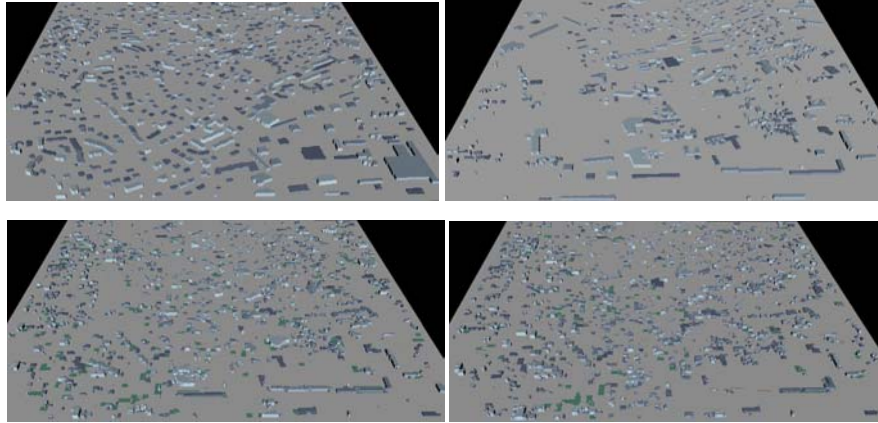


Figure 7: example of pavement-controlled fusion (down) on real cities (Saint Genis and Vénissieux, up)

## 2.4. Shape and Detail Filtering

### *Adjusting IFS Scale for Detail Filtering*

Since IFS coding does not take account for dimensions, it is easy to mathematically rebuild its attractor at any scale. This property leads to what has been called “fractal zoom”, that can be used with values greater than one (creation of fractal detail), or less than one (shape simplification). So we denote a correspondence between the fractal zoom and the generation of continuous “levels of detail” for objects, that can be used within a real-time wandering (fig. 8).

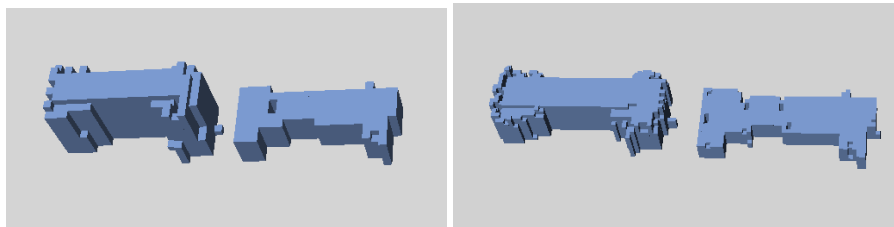


Figure 8 : two versions of the same buildings (before and after a 2x fractal zoom)

### *Shape Filtering within the Domain Blocks Pool*

On the other hand, the encoding of range blocks from domain blocks being surjective, some domain blocks may be used more than other ones, and our experiments confirm this property, that gives an indication on the quantity of

generative information used to approximate an urban fabric. Therefore, our idea was to implement a low-pass filtering on the IFS domain block calling frequencies, estimated for each range block containing the analysed pixel of the image, and then to recalculate the attractor of the IFS. This can lead to drastic geometric simplifications, depending on the value of the cut frequency  $f_c$  (fig. 9).

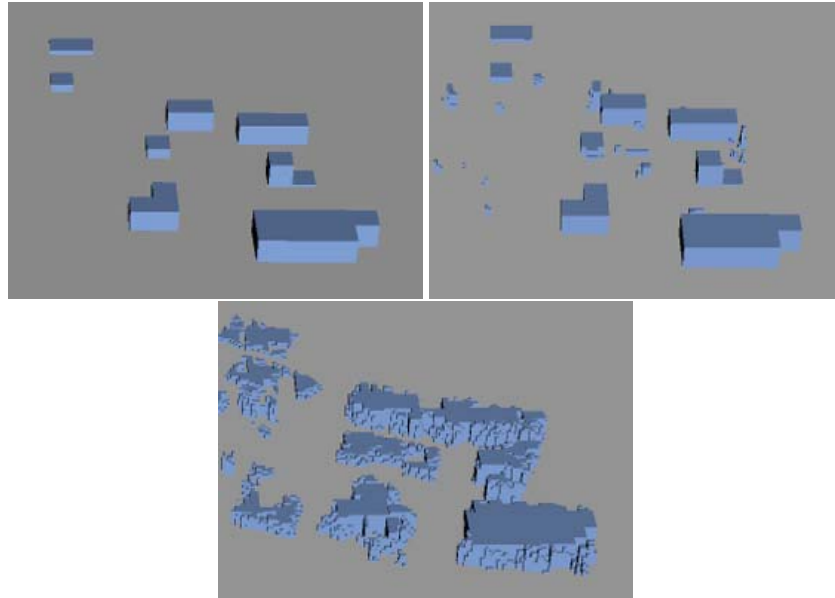


Figure 9 : low-pass filtering with IFS ( $f_c=1$  ;  $f_c=10$  ;  $f_c=100$ )

### 3. FROM GENERATED SCENES TO REAL WORLD

#### 3.1. Towards Urban Shapes Interpretation and Classification

Because IFS do not take account for dimensions, it is first necessary to provide the correspondence between the pixel and grey level units and the size of the objects in the real world. Then, a general method of correspondence between virtual objects and real world ones has been proposed, based on the mixed criterion (surface, height), and allows a primary classification of generated urban objects, completed by some quick shape analysis to help identifying the type of a building, for example. This typology contains 7 types of objects : buildings and houses (blue grey), urban furniture (light orange), trees and vertical vegetation (light green), ground-levelled objects (swimming pools, parkings, lawns, ponds ;

dark green), fountains (dark blue), shelters of garden (brown), electrical posts and public lamps.

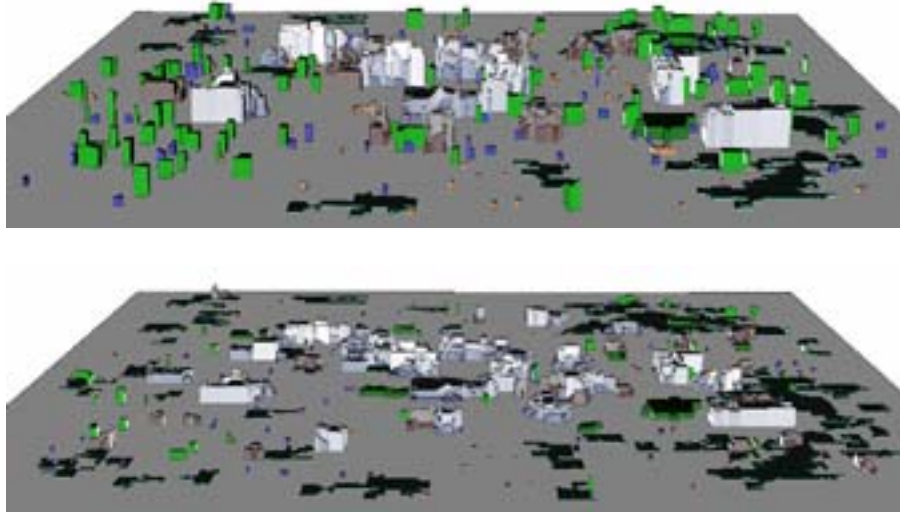


Figure 10 : two examples of classification of urban objects with a colour table

Instead of using raw objects as they are generated, we could operate some substitutions with other ones in typical libraries representative of certain urban atmospheres, for example. But this solution hasn't been already implemented.

### **3.2. Simplifying and Smoothing Generated Shapes**

Some algorithm developments have been required for smoothing irregular distribution of pixels, due to the jaggy appearance of vectorized pixels and the fractal nature of generation, and for obtaining simplified geometric shapes. Our work involves many existing simplification algorithms (Douglas-Peucker, characteristic vertex extraction), combined in a robust approach, introducing the notion of "significant geometric detail", with a scale tolerance factor (Fig. 6).

Another promising way of research concerns local adjustment of pre-defined configurations of "common angles" in architecture, up to global adjustment with constraints (for example, for placing roof shapes).

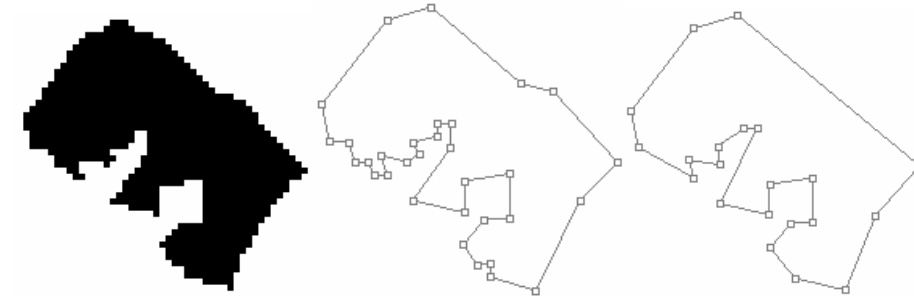


Figure 11 : a noisy building – outline with details – simplified outline (tolerance = 1/20)

### 3.3. Adding Automatic Garbage Dump to the Scenes

Automated generation of streets and places graphs from urban imprints maps is an interesting research topic. This “raster approach” of the problem is rather new comparing to the one dealing with vector objects, and avoids the difficult task of shape vectorisation in noisy environments. Our still progressing work is done in three stages :

- extraction of geometric structuring characteristics from the maps : graphs of streets and places, combined within a technique for spatially grouping houses and buildings in neighbourhoods ;
- geometric generation of corresponding smooth 3D objects in VRML ;
- search for heuristic methods to qualitatively identify plausible elements of garbage dump networks (ex : boulevards, avenues, alleys, water streams).

We’re still working on opened and closed connex graphs of street network. We apply some

« mathematical morphology » basic tools for extracting homotopic skeletons of the ground zones and streets width (fig. 12). We obtain two types of graph, depending on the possibility to connect the city to its environment (opened city), or not (secluded city). To improve the quality of morphological processing, we

work on super scanned images (a factor of 4 seems to be sufficient for 256 x 256 or 512 x 512 images).

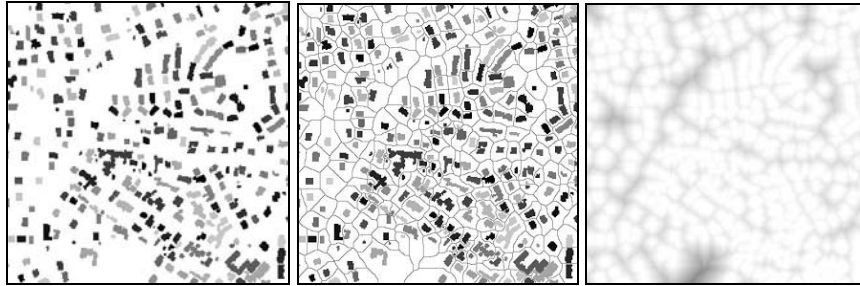


Figure 12 : homotopic opened skeleton (b) of part of Saint Genis city (a) and its street-width map (c)

## 4. CONCLUSIONS AND FUTURE WORKS

### 4.1. Discussion

We have shown in this paper how it is possible to encode simplified  $2D\frac{1}{2}$  city models using an IFS compression technique derived from Jacquin's algorithm. Cutting urban maps with the aid of a square pavement allows local control on the content of the split zones. In this purpose, we initiated a genetic-like approach to share information between IFS coding two (or more) city models, in order to compose new urban and architectural shapes by fusion and mutation. The possibility to deal with real or synthetic urban fabrics opens a wide field of creation, and many ideas have been suggested for that. We mostly obtain orthomorphic geometric models, because of the approach of converting blocks of connex pixels as cubes with their borders. Recent references on city modelling use such objects (Parish, Muller, 2001), where buildings are designed with the aid of L-systems. One can also observe the similitude between the geometric aspect of our models and the famous « architectones » of Kasimir Malevich (Figure 13).

But, up to now, we've just used a uniform and general square pavement. The fact we did not consider other pavements more suitable for the process of isolated buildings or groups of buildings results in discontinuities and loss of topological identity, even if we strove to minimize them in our pavement-based fusion algorithm. Indeed, a related difficulty is the adjustment of the range size parameter  $B$  : if  $B$  is too small, only the outline of the objects are coded by *lifs*, and if  $B$  is too high, it can be very hard to find some blocks similarities. On the



other hand, the IFS approximation quality requires a sufficient resolution for images. These two constraints result in higher computation time for *lifs*, even with the uniform square pavement.

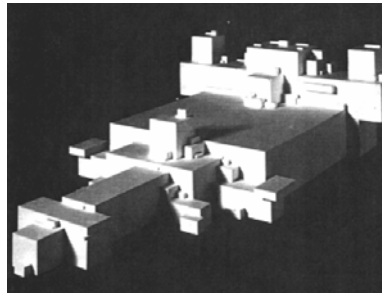


Figure 13 : a famous cubic architecton of Kasimir Malevich (1926)

## 4.2. Remaining Investigations

From a scientific point of view, several ways of research remain :

Our experiments still suffer from a lack of theoretical developments on IFS coding and partitioning for the use of fusion between several city models. It is important to search for better pavements of range and domain blocks, well fitted to match models and process properties.

A semi-synthetic approach for IFS will be explored, in order to reproduce given models of spatial distribution of shapes, using “condensation IFS” that allow the import of extern objects in non coding blocks.

Some experiments with genetic algorithms have to be done to optimise the fusion results, given some shape or statistical distribution criterions extracted from real cities, or by applying the famous “universal distribution law” (Salingaros, 1999).

We also envision the study of location and recombinant mutations in order to increase the size of the IFS original extent, by distributing some *lifs* or groups of *lifs* to other places. From a simulation point of view, this would become a first step towards infinite generation of non-repetitive urban fabric.

## 5. REFERENCES

- Bailly, E. (1998) *Fractal geometry and simulation of urban growth*, UMR CNRS Espace 6012, Nice.
- Barnsley, M. (1993) *Fractal image compression*, AK Peters, Ltd, Wellesley.
- Barnsley, M. (1992) Image coding based on a fractal theory of iterated contractive image transformation, *IEEE transactions on image processing*, 1:18-30.
- Batty, M., Longley, P.A. (1994), *Fractal Cities: A Geometry of Form and Function*, Academic Press, London and San Diego.
- DER (2000), Développement d'un Environnement logiciel de REalité Virtuelle Elaboré, *Projet de recherche DEREVE de la région Rhône-Alpes*, LIGIM, Université Lyon I.
- Frankhauser, P. (1994) *La Fractalité des Structures Urbaines*, Collection Villes, Anthropos, Paris.
- Frankhauser, P. (1997) *L'approche fractale : un nouvel outil de réflexion dans l'analyse spatiale des agglomérations urbaines*, Université de Franche-Comté, Besançon.
- Gentil, C. (1992) *Les fractales en synthèse d'images : le modèle IFS*, Thèse, LIGIM, Université Lyon I, Lyon.
- Hutgen, B., Hain, T. (1994) On the convergence of fractal transforms, *Proceedings of ICASSP*, 561-564.
- Hutchinson, J. (1981) Fractals and self-similarity, *Indiana University Journal of Mathematics*, 30:713-747.
- Jacquin, A.E. (1992) Image coding based on a fractal theory of iterated contractive image transformations, *IEEE transactions on image processing*, 1(1):18-30.
- Makse, H.A. (1996) Modelling fractal cities using the correlated percolation model, *Fractal and granular media conference.*, session C18
- Marsault, X. (2002) Application des Iterated Function Systems (IFS) à la composition de tissus urbains tridimensionnels virtuels, *Autosimilarités et applications*, Cemagref, Clermont Ferrand.
- Parish, Y., Muller, P. (2001) Procedural modelling of cities, *SIGGRAPH*.
- Renders, J.M. (1995) *Algorithmes génétiques et réseaux de neurones*, Editions Hermès.
- Sala, N. (2002) The presence of the self-similarity in architecture : some examples, in M.M.Novak (ed), *Emergent Nature*, World Scientific, 273-283.
- Salingaros, N. (1999) A universal rule for the distribution of sizes, *Environment and Planning B : planning and Design*, 26:909-923, Pion Publications.

- Torrens, P., (2000) *How cellular models of urban systems work*, CASA, Angleterre.
- Vences, L., Rudomin L. (1997) *Genetic algorithms for fractal image and image sequence compression*, Instituto Tecnológico de Estudios Superiores de Monterrey, Camus Estado de Mexico, Computation Visual.
- Wolozyn, P., (1998) *Caractérisation dimensionnelle de la diffusivité des formes architecturales et urbaines*, Thèse, Laboratoire CERMA, NANTES.