

# Parameter identification and algorithmic construction of fractal interpolation functions: Applications in digital imaging and visualization

Polychronis Manousopoulos\*

National and Kapodistrian University of Athens  
Department of Informatics and Telecommunications  
polyman@di.uoa.gr

**Abstract.** This dissertation examines the theory and applications of fractal interpolation. Its main contribution is the parameter identification, algorithmic construction and applications of fractal interpolation. We focus on the self-affine and piecewise self-affine fractal interpolation functions that are based on the theory of iterated function systems. Specifically, we present two novel methods for parameter identification that are based on minimising the symmetric difference between bounding volumes of appropriately chosen points, achieving lower errors compared to existing methods. We also present a novel method that aims at preserving the fractal dimension of the initial set of points. Beyond these, we introduce a new method for curve fitting using fractal interpolation, allowing a more economical representation compared to existing ones. Moreover, we construct non-tensor product bivariate fractal interpolation surfaces. As far as the applications are concerned, we focus on isosurface triangulation, point cloud modelling, active shape models as well as representation and compression of medical and geographic data; fractal interpolation is used as the core of the proposed methods yielding better results or overcoming limitations of existing methodologies.

**Keywords:** fractal interpolation, iterated function system, fractal dimension, curve fitting, bounding volume, vertical scaling factor.

## 1 Introduction

The subject of this dissertation belongs to the greater field of knowledge of fractal geometry and its applications, such as data imaging, computer vision and visualization. Its main contribution is the parameter identification, the algorithmic construction and the applications of fractal interpolation. We focus on self-affine and piecewise self-affine fractal interpolation functions that are based on the theory of iterated function systems.

Fractal interpolation, as defined by M. F. Barnsley and other researchers, is an alternative to traditional interpolation techniques aiming mainly at data

---

\* Dissertation Advisors: Vassileios Drakopoulos, Phd - Theoharis Theoharis, Associate Professor.

which present detail at different scales or some degree of self-similarity. These characteristics imply an irregular, non-smooth structure that is inconvenient to describe e.g. by polynomials. Examples of successful use of fractal interpolation include projections of physical objects such as coastlines and plants, or experimental data of non-integral dimension. Fractal interpolation provides a constructive way to describe data as opposed to the descriptive ways employed by most traditional methods.

The generic problem we intend to solve is the following. Given a set of points, we select a subset of them as interpolation points and construct a fractal function that passes through them. The identification of its parameters, free or not, is important since it determines the quality of interpolation with respect to the initial set of points. Specifically, the closeness of fit of a fractal interpolation function is mainly influenced by the determination of its vertical scaling factors. No direct way to determine the optimum values of these factors exists and most popular approaches employ mainly algebraic or geometric methods. Our motivation is to create an alternative methodology for determining the vertical scaling factors using bounding volumes of appropriately chosen points, such that the resulting fractal function provides a closer fit, with respect to some metric, to the original points. We have developed two such methods for both the self-affine and piecewise self-affine fractal interpolation functions: Bounding rectangles and convex hulls. The first allows the calculation of the optimum vertical scaling factors using analytic expressions, while the second provides tighter bounds and efficient algorithmic calculations. Furthermore, we present a novel method for piecewise self-affine fractal interpolation functions that aims at preserving the fractal dimension of the initial set of points. Results indicate that the proposed methods are effective, achieving lower errors compared to existing ones.

Literature on fractal interpolation focuses mainly on functions, i.e. the data points are linearly ordered with respect to their abscissa and the interpolant is a function of (usually) non-integral dimension. In practice, however, there are many cases where the data are suitable for fractal interpolation but define a curve rather than a function, e.g. when modelling coastlines or plants. So, it is useful to extend fractal interpolation to include curves as well, an issue not fully addressed so far. Existing methods are based on generalizations to higher dimensions, on the use of index coordinates or on non-affine models. We create a method that is more accurate and economical than the existing ones, thus being more suitable for practical applications such as shape representation. Specifically, the proposed method involves applying a reversible transformation to the data points, such that they define a function on the plane. These are then interpolated as usual and the constructed function is transformed back to the original coordinates in order to obtain a curve that interpolates the original points. The proposed method as well as the most popular existing ones are compared in practical applications showing the advantages of the proposed one in terms of either accuracy or compression ratio.

We also examine the case of fractal interpolation surfaces, where ensuring continuity is a more complicated task. Specifically, we construct non-tensor prod-

uct bivariate fractal interpolation surfaces by extending and correcting an existing method. Moreover, we examine the issue of continuity of such surfaces presenting and comparing various approaches.

As far as the applications of fractal interpolation are concerned, we focus on data imaging, visualization and computer vision, as well as data modelling. Specifically, we present (a) a new algorithm for triangulating isosurfaces using fractal interpolation surfaces which is based on the methodology of the well-known marching cubes algorithm, achieving better results while producing an adjustable amount of output primitives, (b) a new method for modelling point clouds based on fractal interpolation, which achieves considerable compression ratios and allows multiresolution reconstruction, (c) active shape models using fractal interpolation, which present the advantage of requiring a considerably smaller number of landmark points and thus a simpler annotation process, and finally (d) the application of fractal interpolation to medical and geographic data, demonstrating its suitability in these cases.

## 2 Fractal interpolation: Parameter identification and algorithmic construction

### 2.1 Fractal interpolation functions

Let  $\Delta_1, \Delta_2$  be partitions of the real compact interval  $I = [a, b]$ , i.e.  $\Delta_1 = \{u_0, u_1, \dots, u_M\}$  satisfying  $a = u_0 < u_1 < \dots < u_M = b$  and  $\Delta_2 = \{x_0, x_1, \dots, x_N\}$  satisfying  $u_0 = x_0 < x_1 < \dots < x_N = u_M$ , such that  $\Delta_1$  is a *refinement* of  $\Delta_2$ . Let us represent as  $P = \{(u_m, v_m) \in I \times \mathbb{R}: m = 0, 1, \dots, M\}$  the given set of *data points* and as  $Q = \{(x_i, y_i) \in I \times \mathbb{R}: i = 0, 1, \dots, N \leq M\}$  a subset of them, the *interpolation points*. The subintervals of  $\Delta_2$  are known as *interpolation intervals* and may be chosen equidistantly or not. The data points within the  $n$ th interpolation interval  $I_n = [x_{n-1}, x_n]$  are represented as  $P_n = \{(u_i, v_i) : x_{n-1} \leq u_i \leq x_n\}$  for all  $n = 1, 2, \dots, N$ .

Let  $\{\mathbb{R}^2; w_n, n = 1, 2, \dots, N\}$  be an *iterated function system*, or *IFS* for short, with affine transformations

$$w_n \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} a_n & 0 \\ c_n & s_n \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} d_n \\ e_n \end{bmatrix}$$

constrained to satisfy  $w_n [x_0 \ y_0]^T = [x_{n-1} \ y_{n-1}]^T$  and  $w_n [x_N \ y_N]^T = [x_n \ y_n]^T$  for every  $n = 1, 2, \dots, N$ . After solving the above equations, the real numbers  $a_n, d_n, c_n, e_n$  are completely determined by the interpolation points; the  $s_n$  are *free* parameters satisfying  $|s_n| < 1$  in order to guarantee that the IFS is *hyperbolic* with respect to an appropriate metric, for every  $n = 1, 2, \dots, N$ . The transformations  $w_n$  are *shear transformations*: line segments parallel to the  $y$ -axis are mapped to line segments parallel to the  $y$ -axis contracted by the factor  $|s_n|$ . For this reason, the  $s_n$  are called *vertical scaling* (or *contractivity*) *factors*. It is well known that the attractor  $G = \bigcup_{n=1}^N w_n(G)$  of the aforementioned IFS is the graph of a continuous function  $f: [x_0, x_N] \rightarrow \mathbb{R}$  that passes through the

interpolation points  $(x_i, y_i)$ , for all  $i = 0, 1, \dots, N$ . This function is called *fractal interpolation function*, or *FIF* for short, corresponding to these points. A *section* is defined as the function values between interpolation points. This is a *self-affine* function since each affine transformation  $w_n$  maps the entire (graph of the) function to each section.

The aforementioned model of self-affine FIFs can be extended as follows. Each interpolation interval is associated with a pair of data points called *address points*. Specifically, the interpolation interval  $I_n = [x_{n-1}, x_n]$  is associated with the points  $(x'_{n,1}, y'_{n,1})$  and  $(x'_{n,2}, y'_{n,2})$  for  $n = 1, 2, \dots, N$ , where  $(x'_{n,k}, y'_{n,k}) = (u_m, v_m)$  for all  $k \in \{1, 2\}$  and some  $m = 0, 1, \dots, M$ . Each pair of address points defines the *address interval*  $[x'_{n,1}, x'_{n,2}]$  and we have  $x'_{n,1} < x'_{n,2}$  for all  $n = 1, 2, \dots, N$  by definition. The set of data points within each address interval is denoted as  $P^{[n]} = \{(u, v) \in P : x'_{n,1} \leq u \leq x'_{n,2}\}$ , for all  $n = 1, 2, \dots, N$ . The affine transformations  $w_n$ ,  $n = 1, 2, \dots, N$  are now constrained to satisfy  $w_n [x'_{n,1} \ y'_{n,1}]^T = [x_{n-1} \ y_{n-1}]^T$  and  $w_n [x'_{n,2} \ y'_{n,2}]^T = [x_n \ y_n]^T$  for all  $n = 1, 2, \dots, N$ , i.e. each address interval is mapped to its corresponding interpolation interval. After solving the above equations, the real numbers  $a_n, d_n, c_n, e_n$  are completely determined by the interpolation and address points; the vertical scaling factors  $s_n$  are again free parameters satisfying  $|s_n| < 1$ . Let  $W(A) = \bigcup_{n=1}^N w_n(A^{[n]})$ , where  $A \in \mathcal{H}(\mathbb{R}^2)$  and  $A^{[n]} = \{(x, y) \in A : x'_{n,1} \leq x \leq x'_{n,2}\}$ , for  $n = 1, 2, \dots, N$ . The unique set  $G \equiv A_\infty = \lim_{k \rightarrow \infty} W^k(A_0)$ , for all  $A_0 \in \mathcal{H}(\mathbb{R}^2)$ , is the graph of a continuous function  $f: [x_0, x_N] \rightarrow \mathbb{R}$  that passes through the interpolation points  $(x_i, y_i)$ , for all  $i = 0, 1, \dots, N$ . This function is called *piecewise self-affine FIF*, since each transformation  $w_n$  maps the part of the (graph of the) function defined by the corresponding address interval to each section.

Although a (piecewise) self-affine or FIF passes by definition through its interpolation points, this is not necessarily the case for the remaining data points  $P \setminus Q$ . The closeness of fit depends solely on each  $s_n$ ,  $n = 1, 2, \dots, N$ , the only free parameters for a given  $P$ , and can be measured e.g. as the squared error between the ordinates of the original and the reconstructed points, as the Hausdorff distance  $h(P, G)$ , or as the *Modified Hausdorff Distance*  $h_{MHD}(P, G) = \max \{1/|P| \sum_{a \in P} \min_{b \in G} \|a - b\|, 1/|G| \sum_{b \in G} \min_{a \in P} \|a - b\|\}$ , where  $|\cdot|$  denotes the cardinality of a set.

In [2], [6] and [7] we propose to work with *bounding volumes* of  $P^{[n]}$  ( $= P$  for the self-affine case) and  $P_n$  in order for the transformed points  $w_n(P^{[n]})$  to best approximate the data points within  $P_n$ . Let  $B^{[n]} \in \mathcal{K}_0^2 (= B$  for the self-affine case) be a bounding volume of  $P^{[n]}$ , where  $\mathcal{K}_0^2$  denotes the set of convex, compact subsets of  $\mathbb{R}^2$  with non-empty interior, and  $B_n \in \mathcal{K}_0^2$  be convex bounding volumes of  $P_n$  for every  $n = 1, 2, \dots, N$ . In other words, it is  $P^{[n]} \subset B^{[n]}$  and  $P_n \subset B_n$ , for every  $n = 1, 2, \dots, N$ . We use the *symmetric difference metric*

$$\delta^S(K, L) = \mathcal{H}^2(K \triangle L) = \mathcal{H}^2((K \setminus L) \cup (L \setminus K)), \quad K, L \in \mathcal{K}_0^2 \quad (1)$$

where  $\mathcal{H}^2$  denotes the Hausdorff measure in  $\mathbb{R}^2$ , in order to minimize the area of the symmetric difference  $B_n \triangle w_n(B^{[n]})$ ,  $n = 1, 2, \dots, N$ . Notice that since we are constrained in  $\mathcal{K}_0^2$  the Hausdorff measure coincides with the Lebesgue

measure, i.e. the area, in  $\mathbb{R}^2$ . So, Eq. (1) can be written in the form

$$\delta^S(K, L) = \text{area}(K \setminus L) + \text{area}(L \setminus K) = \text{area}(K \cup L) - \text{area}(K \cap L). \quad (2)$$

Therefore, by selecting the values of  $s_n$  that result in the maximum overlap of the respective bounding volumes we are able to produce a better approximation of the data points. This approach has the advantage that, for suitably chosen bounding volumes  $B^{[n]}$  and  $B_n$ , we are able to efficiently obtain the optimum  $s_n$  using either analytic expressions or efficient algorithms. Two types of bounding volume for the minimization of  $\delta^S(B_n, w_n(B^{[n]}))$ , for all  $n = 1, 2, \dots, N$  are selected, namely the *bounding rectangle* and the *convex hull*. The first type allows the calculation of the optimum vertical scaling factors using analytic expressions, while the second provides tighter bounds and efficient algorithmic calculations.

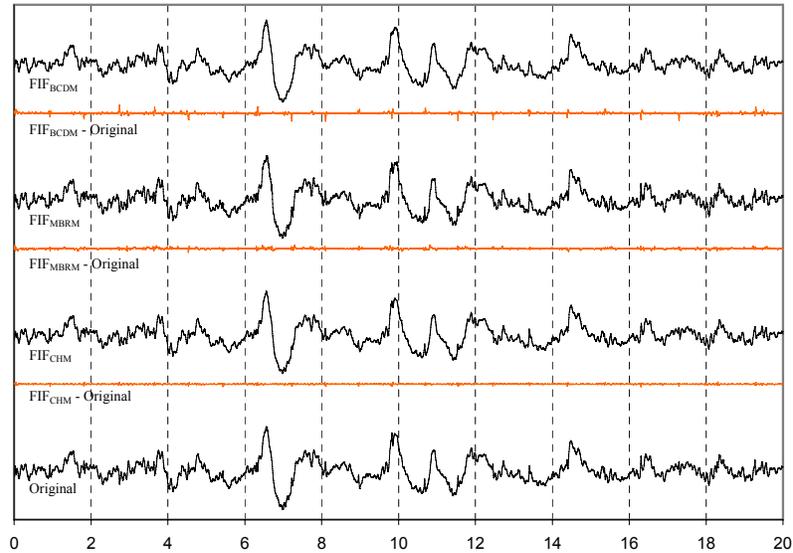
The *Minimum Bounding Rectangle Method*, or *MBRM* for short, employs bounding rectangles aligned with the axes of the co-ordinate system. Let  $R_n$  be the MBR of  $P_n$  and  $R^{[n]}$  ( $= R$  for the self-affine case) the MBR of  $P^{[n]}$ . In view of Eq. (1), our aim is the minimization of  $\delta^S(R_n, w_n(R^{[n]}))$ , for every  $n = 1, 2, \dots, N$ . This is achieved by minimizing the area of the non-overlapping parts of  $R_n$  and  $w_n(R^{[n]})$ . The possible cases of intersection of  $R_n$  and  $w_n(R^{[n]})$  along with the analytic expressions for the optimum  $s_n$  that minimize the area of the non-overlapping parts in each case are presented in detail in [6].

The *Convex Hull Method*, or *CHM* for short, employs the convex hull as bounding volume. This provides a tighter bound than the rectangle and is actually the smallest convex set containing the data points. Similarly to the case of bounding rectangles, we want to minimize the area of the nonoverlapping parts of the convex hull of the points in the  $n$ th interpolation interval and the transformation of the convex hull of  $P^{[n]}$  under  $w_n$ . According to Eq. (2), this is

$$\begin{aligned} \delta^S(CH(P_n), CH(w_n(P^{[n]}))) &= \text{Area}\{CH(P_n)\} + \text{Area}\{CH(w_n(P^{[n]}))\} - \\ &\quad - 2\text{Area}\{CH(P_n \cap w_n(P^{[n]}))\}, \end{aligned} \quad (3)$$

where  $CH(\cdot)$  is the convex hull of a set of points. The calculation of the optimum  $\hat{s}_n$  cannot be performed analytically as in the MBRM. As implied by Eq. (3), the calculation of  $\delta^S$  is algorithmic and involves the computation of convex hulls, polygon intersections and areas. As we suggest in [6], a method for one-dimensional minimization without derivatives should be used, such as R. P. Brent's method which is a *bracketing method* with parabolic interpolation.

Also, in [7] we propose an alternative approach for piecewise self-affine FIFs that aims at preserving the box-counting dimension of the original data points. It is known that the box-counting dimension  $D$  of a piecewise self-affine FIF satisfies under certain conditions the equation  $\rho(CS(D)) = 1$ , where  $\rho(\cdot)$  is the spectral radius,  $C = (c_{ij})$  is the connection matrix and  $S(d) = \text{diag}\{|s_1||a_1|^{d-1}, \dots, |s_N||a_N|^{d-1}\}$ . We examine the case when, additionally for all  $n = 1, 2, \dots, N$ , we have that (a) the vertical scaling factors are the same, i.e.  $s_n = s$ , (b) the address intervals are of equal length, i.e.  $x'_{n,2} - x'_{n,1} = L'$  and  $\sum_{j=1}^N c_{ij} = c$ , for every  $i = 1, 2, \dots, N$ , (c) the interpolation intervals are of equal length, i.e.



**Fig. 1.** The original signal (bottom), its reconstruction by the CHM and their difference (2<sup>nd</sup> from bottom), its reconstruction by the MBRM and their difference (3<sup>rd</sup> from bottom), and its reconstruction by the BCDM and their difference (top).

$x_n - x_{n-1} = L$ , so  $a_n = L/L' = a$ . As we prove in [7], if the vertical scaling factors are calculated as  $|s| = 1/(c|a|^{D-1})$ , then the box-counting dimension of the resulting piecewise self-affine FIF is  $D$ , which therefore can be set equal to the dimension of the set of points  $P$  in order to preserve it. The sign of the vertical scaling factor is determined by selecting the one that minimizes the Hausdorff distance between the original and the reconstructed points. Henceforth, we will call this method the *Box-Counting Dimension Method*, or *BCDM* for short.

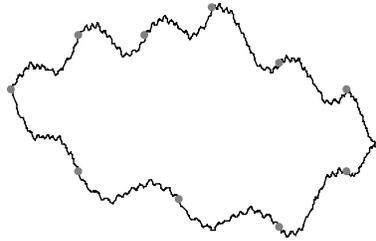
The lower part of Figure 1 presents a part of an EEG signal<sup>1</sup> consisting of 5000 points. In the same figure, the piecewise self-affine FIFs constructed by the CHM, the MBRM and the BCDM are shown from bottom to top. Interpolation intervals of 25 points and address intervals of 500 points have been used. The address intervals have been chosen to be consecutive and non-overlapping, while the optimum one for each interpolation interval has been chosen in terms of the Hausdorff distance between the original and the reconstructed points. Under each graph, its difference from the original is also depicted in red. As can be seen from the figure, the reconstructed functions interpolate the data points quite successfully despite the sparsity of the interpolation points. Detailed comparison of the proposed methods along with the popular geometric and the algebraic ones of D. S. Mazel and M. H. Hayes indicates that MBRM and CHM outperform both existing methods in terms of Hausdorff distance and Modified Hausdorff

<sup>1</sup> The data are part of the MIT-BIH Polysomnographic Database.

distance between the original and the reconstructed data points. The BCDM produces acceptable results even though it does not aim at minimizing some error measure, while preserving the dimension of the data.

## 2.2 Fractal interpolation curves

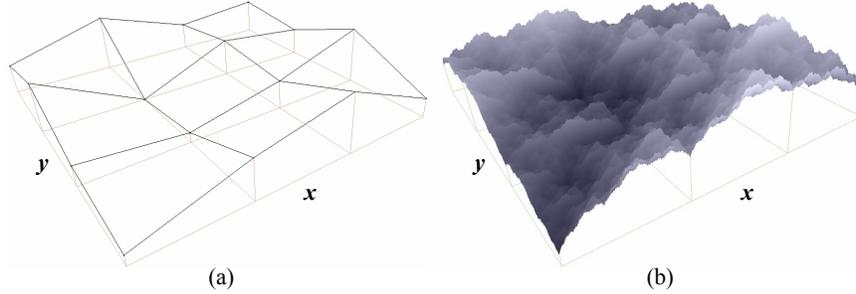
Fractal interpolation literature focuses on functions, i.e. the data points are linearly ordered with respect to their abscissa. In practice, however, there are cases when the data are suitable for fractal interpolation but define a curve rather than a function, e.g. when modelling coastlines or plants. In order to model curves by fractal interpolation, existing methods use generalizations to higher dimensions, index coordinates or non-affine models. In [4] and [9] we propose a novel method for representing curves using fractal interpolation. The main advantage of the proposed method is offering a more economical representation of the data compared to existing methods, thus allowing higher compression ratios. The central idea is to apply a reversible transformation to the data points such that they define a function on the plane. A FIF is constructed as usual and then it is transformed back to the original coordinates in order to obtain a curve that interpolates the original points. An example is depicted in Figure 2, where a fractal interpolation curve is constructed for a set of 11 interpolation points.



**Fig. 2.** A fractal interpolation curve constructed by the proposed method.

## 2.3 Fractal interpolation surfaces

A *fractal interpolation surface* is a function that belongs in the three-dimensional space and has Hausdorff-Besicovitch dimension between 2 and 3; it is defined for sets of interpolation points formulated as  $\{(x_i, y_j, z_{ij} = z(x_i, y_j)) \in \mathbb{R}^3, i = 0, 1, \dots, M; j = 0, 1, \dots, N\}$ . Such sets of data points usually form triangular or rectangular grids. The key difficulty in the construction of fractal interpolation surfaces is ensuring their continuity. In [1] we construct non-tensor product bivariate fractal interpolation surfaces for arbitrary interpolation points and vertical scaling factors. This is an important result, since most existing methods



**Fig. 3.** (a) The rectangular lattice. (b) The fractal interpolation surface.

impose restrictions on the interpolation points or the factors in order to achieve continuity. In Figure 3(a) a rectangular lattice is depicted; in Figure 3(b) the respective fractal interpolation surface constructed by the proposed method using vertical scaling factors  $s_{m,n} = 0.45$  for all  $m, n = 1, 2, 3$  is depicted.

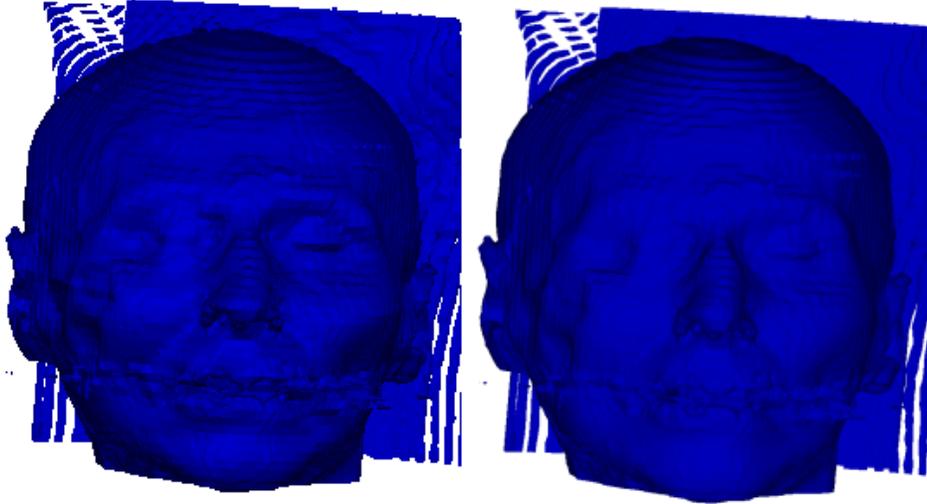
### 3 Applications of fractal interpolation

#### 3.1 Volume data visualization

Visualization of medical or experimental 3D data is often achieved by extracting an intermediate geometric representation of the data. One such popular method for extracting an isosurface from volume data is the *Marching Cubes (MC)* algorithm, which creates a polygon mesh by sampling the data at the vertices of the cubes of a 3D grid. In [5] we present a novel method for isosurface approximation that uses the vertex extraction phase of the MC algorithm, but subsequently represents the data by fractal interpolation surfaces instead of a polygon mesh, considering additionally the surface topology in the interior of the cubes. The proposed *Fractal Marching Cubes (FMC)* method is appropriate for isosurfaces that are not locally flat, such as natural structures. Another advantage is that a coarser grid resolution can typically be used, since fractal interpolation surfaces are particularly good at representing detailed, irregular or self-similar structures. The multiresolution extension of the method is also straightforward. In Figure 4, an example of medical data visualization is presented, showing that the proposed FMC method can achieve superior results compared to the original MC.

#### 3.2 Image understanding

*Active Shape Models (ASM)* are used in image understanding either for locating shapes in static images or for motion tracking in image sequences. They have been successfully used in practice, especially in biomedical applications. ASM use a statistical representation of shapes. The shape in each image of a training

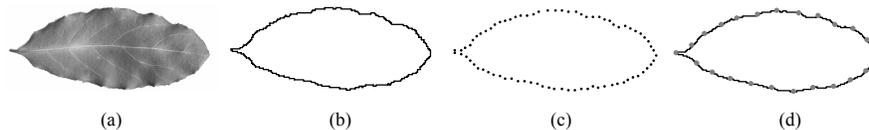


**Fig. 4.** CT visualization of tissue using the MC algorithm and grid resolution  $128 \times 128 \times 52$ .(left) CT visualization of tissue using the FMC algorithm and grid resolution  $128 \times 128 \times 52$ . (right)

sample is annotated by a set of landmark points and is thus represented as the vector of the point coordinates. The shape vectors are aligned and then described by applying Principal Component Analysis (PCA) to them. In this way, we have a mean shape and the possible variation along the eigenvectors of the covariance matrix, thus defining the space of allowable shapes implied by the sample. It is then possible to locate allowable shapes in new images, by an iterative image search algorithm that attracts an initial shape approximation towards the image edges while remaining inside the allowable space.

In many ASM applications a significant number of landmark points or training images may be required, thus rendering the (partly or wholly manual) annotation process time-consuming and error-prone. In [3] we introduce the *Fractal Active Shape Models (FASM)* aiming at considerably reducing the number of required landmark points. They describe a shape by fractal interpolation curves and represent it by the vector of the interpolating IFS transformation coefficients. Afterwards, PCA is applied to the shape vectors in order to define the allowable shape space, while image search is performed in a similar manner. In Figure 5(b), the shape boundary of Figure 5(a) is depicted. The shape can be represented either by the 77 landmark points of Figure 5(c) or by the fractal interpolation curve of 23 interpolation points of Figure 5(d). In both cases the representation quality is similar, but the proposed method requires about 30% of the ASM landmark points. The feasibility of this method is based on the fact that there is a continuous dependence between the transformations' coefficients and the attractor of an IFS, while its efficiency in image search has been

shown in practice. It is especially useful for image samples containing irregular, non-smooth shapes.



**Fig. 5.** (a) A leaf of the plant *Laurus nobilis*. (b) The shape boundary consisting of 930 points. (c) The 77 landmark points. (d) The fractal interpolation curve of 23 points.

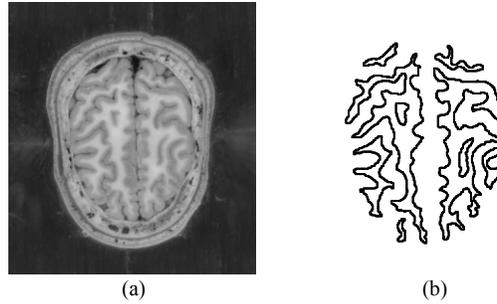
### 3.3 Point cloud modelling

Point clouds are used for modelling 3D objects, especially those digitized by 3D scanners. In such applications high-resolution digitization is performed resulting in an increased amount of points, e.g. data sets of  $10^7$  points are common in practice. Moreover, additional information, such as normal vector or colour, necessary for photorealistic rendering may be stored. All these result in an increased volume of data, often difficult and time consuming to handle, thus leading to the development of point cloud modelling and compression methodologies.

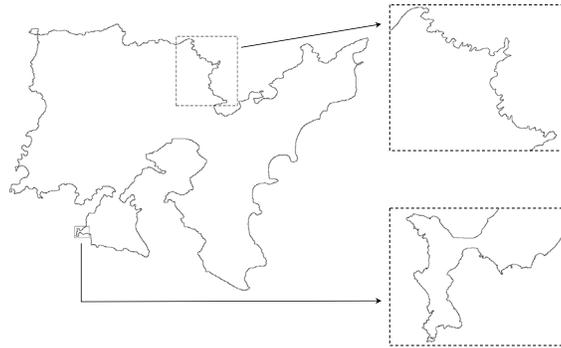
In [8] we propose a novel method based on fractal interpolation for modelling and compressing point clouds. Our aim is (a) to achieve considerable compression ratios and (b) to allow multi-resolution reconstruction, i.e. the ability of a fast approximative reconstruction as well as a slower but accurate one. Each 3D point of the cloud is allowed to be associated with additional coordinates, corresponding e.g. to normal vector or colour. The first step is to partition the point cloud into subsets using the  $k$ -means clustering algorithm. The points of each cluster are sorted in order of mutual proximity and are subsequently represented by multiple FIFs, each one constructed for one of the point coordinates and an appended index coordinate. Every cluster is then modelled by the transformation coefficients of the interpolating IFSs and the point cloud by the coefficients of all clusters. The compression ratio is determined by the density of the interpolation points which defines the number of transformations of each IFS. The reconstruction of the point cloud is achieved by constructing the attractors of all IFSs and subsequently merge them in a single set based on the discardable index coordinate. The cardinality of the reconstructed point cloud can be parametrized through the attractor calculations, thus determining the accuracy and speed of the reconstruction.

### 3.4 Modelling and compression of geographic and medical data

In [7], [4] and [9] we present various applications of fractal interpolation to medical and geographic data. Here we give two such examples. In Figure 6 an example



**Fig. 6.** (a) The brain axial anatomical image. (b) The representation of the white matter using fractal interpolation curves.



**Fig. 7.** The coastline representation at multiple scales.

of applying fractal interpolation to axial anatomical images is presented<sup>2</sup>. Specifically, the boundary of the brain white matter is modelled by multiple fractal interpolation curves constructed by the proposed method using interpolation intervals of 20 points. We note that the usefulness of fractal interpolation in medical applications can be significant, since it has been shown that it can be used for dimension based diagnosis. In Figure 7 a coastline representation is depicted. In this example, the ability of fractal interpolation to represent detail at different scales is emphasized.

#### 4 Conclusions and future work

This dissertation has examined the theory and applications of fractal interpolation contributing to its parameter identification, algorithmic construction

<sup>2</sup> Data from the “Visible Human Project” of the National Library of Medicine.

and applications, such as data imaging, computer vision and visualization. We have presented two novel methods for parameter identification of self-affine and piecewise self-affine fractal interpolation functions; by minimising the symmetric difference between bounding volumes of appropriately chosen points, we have achieved lower errors compared to existing methods. Furthermore, a novel method that aims at preserving the fractal dimension of the initial set of points has been presented. Beyond these, we have proposed a new method for curve fitting using fractal interpolation, allowing a more economical representation than existing methods. Moreover, we have constructed non-tensor product bivariate fractal interpolation surfaces. As far as the applications are concerned, we have focused on isosurface triangulation, point cloud modelling, active shape models, and representation and compression of medical and geographic data. In all applications, fractal interpolation has been used as the core of the proposed methods yielding better results or overcoming limitations of existing methodologies.

Future research directions can be derived based on each individual problem studied in the context of this dissertation. Extending the proposed parameter identification methods to fractal interpolation surfaces, as well as achieving automated diagnosis in medical applications using fractal interpolation models are representative examples of such further work.

## References

1. V. Drakopoulos and P. Manousopoulos. Non-tensor product bivariate fractal interpolation surfaces on rectangular grids. *Submitted*, 2009.
2. V. Drakopoulos and P. Manousopoulos. One dimensional fractal interpolation: Determination of the vertical scaling factors using convex hulls. In *Topics on Chaotic Systems: Selected Papers from Chaos 2008 International Conference*, 2009.
3. P. Manousopoulos, V. Drakopoulos, and T. Theoharis. Fractal Active Shape Models. In *Computer analysis of images and patterns, 12th international conference*, volume 4673 of *Lecture Notes in Computer Science*, pages 645–652. Springer, 2007.
4. P. Manousopoulos, V. Drakopoulos, and T. Theoharis. Curve fitting by fractal interpolation. *Transactions on Computational Science*, 1:85–103, 2008.
5. P. Manousopoulos, V. Drakopoulos, and T. Theoharis. Volume data visualization using fractal interpolation surfaces. In *Proceedings of Eurographics 2008 - Short Papers Annex*, pages 287–290, 2008.
6. P. Manousopoulos, V. Drakopoulos, and T. Theoharis. Parameter identification of 1D fractal interpolation functions using bounding volumes. *Journal of Computational and Applied Mathematics*, 233(4):1063–1082, 2009.
7. P. Manousopoulos, V. Drakopoulos, and T. Theoharis. Parameter identification of 1D recurrent fractal interpolation functions with applications to medical data. *Submitted*, 2009.
8. P. Manousopoulos, V. Drakopoulos, and T. Theoharis. Point cloud modeling using fractal interpolation. In *Proceedings of the Chaos 2009 International Conference*, CD version, 2009.
9. P. Manousopoulos, V. Drakopoulos, T. Theoharis, and P. Stavrou. Effective representation of 2D and 3D data using fractal interpolation. In *Proceedings of International Conference on Cyberworlds*, pages 457–464. IEEE Computer Society, 2007.