Laboratório VISGRAF
Instituto de Matemática Pura e Aplicada

Improved Partition of Unity Implicit Surface Reconstruction

Boris Mederos, Sueni Arouca
Marcos Lage, Helio Lopes, Luiz Velho

Technical Report TR-04-06 Relatório Técnico

November - 2006 - Novembro

Improved Partition of Unity Implicit Surface Reconstruction

Boris Mederos
Laboratório Visgraf
Instituto de Matemática Pura e Aplicada
boris@impa.br
Sueni Arouca, Marcos Lage and Hélio Lopes
Laboratório Matmídia – Departamento de Matemática
Pontifícia Universidade Católica do Rio de Janeiro
{sueni,mlage,lopes}@mat.puc-rio.br
Luiz Velho
Laboratório Visgraf
Instituto de Matemática Pura e Aplicada
lvelho@impa.br

Abstract

We consider the problem of infer a shape of a geometric object from a finite set of sample points with normals. Our contribution is to present a new scheme for Implicit Surface Reconstruction based on a domain decomposition technique known as multilevel partition of unity (MPU) which hierarchically divides the domain in parts obtaining local approximation for the object on each part, and then patch all together obtaining a global description of the object. Our new scheme uses ridge regression and weighted gradient one fitting techniques to get better stability on local approximations. The method behave reasonably on sparse set of points and data with holes as those which comes from the 3D scanning of real objects.

Keywords: Implicit modeling, Surfaces reconstruction, Partition of unity.

1 Introduction

The surface reconstruction is playing an important role in computer vision [17] and computer graphics [13]. Reconstruction is a very complex problem not only because the adjacency and proximity relations of data are unknown, but also because there are a lot adversities that need to be faced. The data in which these algorithms are applied come from 3D scanner. Currently this devices are able to handle real objects with increasing complexity and the resulting point clouds of the 3D scanning contain fi ne details, rapidly geometric variations, complex topology and sharp features. Also, the process of capturing the point cloud introduces sparse samples, holes (due to the occlusion of certain part of the object by other ones) and noise.

Several techniques have been studied to solve the surface reconstruction problem along the last two decades producing a variety of algorithms. Several methods are based on Delaunay triangulation concepts [3, 4, 5] and for noisy data [10, 14], other solutions are based on local parameterisations of shape as the Moving Least Square (MLS) approach [12, 1, 2]. Finally, there is also an important class of methods based on implicit function approximations for example, the radial based function (RBF) technique [9, 16, 19] and domain decomposition schema [15].

In this work we will focus on an implicit based scheme for surface reconstruction. Implicit Surface is a very useful representations for 3D objects, mainly because the inferred shape is computed by a formula which allows the computation of basic modeling operations in a relatively easy way. Most of the boundary of man made objects are compose of several patches which can be approximate by algebraic functions [17]. When its shape is complex, a common procedure is to elevate the polynomial degree in order to obtain more precision on the approximation. However, in this case due to the ill poseness of this fitting some spurious connected components appears on the reconstructed surface. An alternative solution is to decompose the domain hierarchically in compact parts and obtain local approximation for the object in each part, and then patch all together in order to obtain a global description of the object. A practical scheme that uses such solution is the method of multilevel partition of unity implicit (MPU) [15]. It provides an adaptive error-controlled approximation of the signed distance function from the surface.

Given a finite set of points $\mathcal{P}$ in $\mathbb{R}^3$ sampled from a surface $\mathcal{S}$ in the space, also assume that for each point $p \in \mathcal{P}$ the unitary normal vector n to the surface at p is given. The objective of this work is to obtain an implicit function $F : \mathbb{R}^3 \to \mathbb{R}$ such that the level sets $F^{-1}(0)$ approximates adaptively $\mathcal{S}$ using local error-control.

The reconstruction scheme proposed in this paper is an extensions to 3D case of surface reconstruction of a recent work [6] proposed to tackle the 2D case of curve reconstructions. In this extension we proposed a weighted local approximations which produced a substantially improvement compared to the direct extension of the two dimensional method. The algorithms is based on the main ideas of MPU [15] to produce a hierarchically subdivision of the domain in severals parts and later computes local shape approximations on each part. Our method presents a different strategy for the local shape approximations and provides the numerical stability of the local approximation by the use of the ridge regression technique and weighted gradient one fitting. As a consequence, it avoids the generation of spurious connected components on the reconstructed implicit surface.

Section 2 introduces some basic concepts, describes the previous and related works in implicit surface fitting and the partition of unity concepts. Section 3 introduces the algorithms and implementation details, Section 4 shows some results. Finally, section 5 concludes and propose several future directions of this work.

2 Implicit surface fitting and partition of unity

2.1 Algebraic

A subset $\mathcal{O} \subset \mathbb{R}^3$ is called an *Implicit Surface* if there is a function $F : U \to \mathbb{R}$, $\mathcal{O} \subset U$, and a real number $c \in \mathbb{R}$ such that $\mathcal{O} = F^{-1}(c)$. The implicit surface $F^{-1}(c)$ is *regular* if F is differentiable and satisfies the condition that at each point $\mathbf{x} \in F^{-1}(c))$ the gradient of F at x does not vanish.

A *polynomial of degree* d defined on the $\mathbb{R}^3$ is a function $P_d : \mathbb{R}^3 \to \mathbb{R}$ given by the following expression:

$$P_d(x, y, z) = \sum_{0 \leq i+j+k \leq d} a_{i,j,k} x^i y^j z^k.$$

An *algebraic surface of degree* d is the implicit surface $P_d^{-1}(0)$.

It is convenient to adopt a suitable notation for P_d. We adopted the vectorial one proposed by Tasdizen et al. in [17], which is:

$$P_d(x, y, z) = \mathbf{v}^t \mathbf{a}, \tag{1}$$

where

$$\mathbf{a} = \begin{bmatrix} a_{0,0,0} & \dots & a_{d,0,0} & a_{0,1,0} & \dots & a_{d-1,1,0} & a_{0,2,0} & \dots & a_{0,d,0} & a_{0,0,1} & \dots & a_{0,0,d} \end{bmatrix}^t$$

and

$$\mathbf{v} = \begin{bmatrix} 1 & \dots & x^d & y & \dots & x^{d-1}y & y^2 \dots y^d & z \dots z^d \end{bmatrix}^t. \tag{2}$$

The elements of the vector $\mathbf{a} \in \mathbb{R}^l$ are the coeffi cients $a_{i,j,k}$ ($0 \leq i$, $0 \leq j$, $0 \leq k$, and $0 \leq i+j+k \leq d$) of P_d, and the elements of the vector $\mathbf{v} \in \mathbb{R}^l$ are the monomials of P_d. The dimension l of the vectors $\mathbf{a}$ and $\mathbf{w}$ depends on the degree d and is obtained by the following expression: $l = \frac{(d+1)(d+2)(d+3)}{6}$.

2.2 Implicit surface fitting

Follows the description of some methods to obtain an implicit surface approximation.

Minimizing the algebraic distance. Suppose that a set of q points $\mathcal{P} = \{\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_q\} \in \mathbb{R}^3$ sampled from a 3D surface $\mathcal{S}$ is given. A simple way to obtain an algebraic surface $P_d^{-1}(0)$ that approximates $\mathcal{S}$ is to minimize the total sum of the squared algebraic distance, denoted by e_{total}, from each point $\mathbf{p}_i \in \mathcal{P}$ to the curve $P_d^{-1}(0)$. Using the vectorial representation of P_d in (1), we can write e_{total} as:

$$e_{total} = \mathbf{a}^t(\sum_{i=1}^{q} \mathbf{v}_i \mathbf{v}_i^t)\mathbf{a},$$

where $\mathbf{v}_i$ corresponds to the evaluation of the vector $\mathbf{v}$, see equation (2), at the points $p_i = (x_i, y_i, z_i)$. This minimization is subject to the constrain that the sum of the squared coeffi cients of P_d is equal to one ($||a||^2 = 1$). Note that without this constrains the minimum of e_{total} is reached at the null vector. A more robust approximation measure is to consider the weighted algebraic distance

$$e_{total} = \frac{\mathbf{a}^t(\sum_{i=1}^{q} w_i \mathbf{v}_i \mathbf{v}_i^t)\mathbf{a}}{\sum_i^q w_i},$$

where $w_i = w(p_i) \geq 0$ is a set of weight and $w(x, y, z)$ is a positive weight function such that $w(x)$ approaches to zero when $\|x\|$ goes toinfi nity or a compactly support function. To improve the notation, defi ne the matrices $\mathbf{M}$ of size $l \times q$ and $\mathbf{S}$ of size $l \times l$ as follow:

$$\mathbf{M} = \begin{bmatrix} \mathbf{v}_1 & \mathbf{v}_2 & \ldots & \mathbf{v}_q \end{bmatrix},$$

and

$$\mathbf{S} = \mathbf{M}\mathbf{M}^t = \frac{\sum_{i=1}^{q} w_i \mathbf{v}_i \mathbf{v}_i^t}{\sum w_i}.$$

Thus, the minimization problem we have to solve is the following:

$$min_{\mathbf{a}}\{\mathbf{a}^t\mathbf{S}\mathbf{a}\},$$
$$\text{subject to:} ||\mathbf{a}||^2 = 1.$$

By the use of the Lagrange multiplier λ, the constrained minimization problem is rewritten as:

$$min_{\mathbf{a}}\{\mathbf{a}^t\mathbf{S}\mathbf{a} + \lambda(\mathbf{a}^t\mathbf{a} - 1)\}.$$

Its solution is the unitary eigenvector of S associated to the eigenvalue of S with smallest value[18].

Although this methods is invariant to affi ne transformations [18], it has some delicate problems. Its results are sensitive to small perturbations on the input data. Moreover, the algebraic surface $P_d^{-1}(0)$ doesn't consider the points continuity. Thus, it could generate undesired connected components or glue components that are originally separated. For more details of this method see [8, 17, 18].

Gradient one fitting. To avoid the problems of continuity and of sensitivity caused by small perturbation on the data input [8] proposed a new method that considers not only the set $\mathcal{P}$ but also a given set of normal unitary vectors $\mathcal{N} = \{\mathbf{n}_1, \mathbf{n}_2, \ldots, \mathbf{n}_q\}$, where the vector n_i is the normal vector to the surface $\mathcal{S}$ at the sampled point $p_i \in \mathcal{P}$.

The gradient of P_d at $p_i = (x_i, y_i, z_i)$, $\nabla P_d(x_i, y_i, z_i) = \left(\frac{\partial P_d}{\partial x}(x_i, y_i, z_i), \frac{\partial P_d}{\partial y}(x_i, y_i, z_i), \frac{\partial P_d}{\partial z}(x_i, y_i, z_i)\right)$, will be used by this method to approximate the direction of the given normal and the tangent vectors. It is important to notice that if the gradient of P_d at $p_i = (x_i, y_i, z_i)$ is not equal zero, then it is perpendicular to the tangent plane of the level surface P_d that pass through the point p_i.

The *gradient one fitting* (GOF) method [8] is in fact an unconstrained least square problem, we have introduced a modifi cation in the original GOF method which consist of a weighted least square problem as follows

$$e_{grad} = \sum_{i=1}^{q} w_i[(P_d(x_i, y_i, z_i)^2 + \mu(\mathbf{n}_i^t \nabla P_d(x_i, y_i, z_i) - 1)^2] \tag{3}$$

where μ is the weight given to the terms that approximates the normals and w_i is a set of weight depending of each p_i. In order to follow the vectorial representation of P_d, we defi ne the following matrices and vectors:

- The matrix $\mathbf{D}_i = \left[\begin{array}{ccc} \frac{\partial \mathbf{v}_i}{\partial x} & \frac{\partial \mathbf{v}_i}{\partial y} & \frac{\partial \mathbf{v}_i}{\partial z} \end{array}\right]$ of size $l \times 3$:
- The gradient vector ∇P_d:
$$\nabla P_d = \nabla(\mathbf{v}_i^t \mathbf{a}) = (\mathbf{D}_i)^t \mathbf{a}.$$
- The matrix $\mathbf{S_N}$ of size $l \times l$:
$$\mathbf{S_N} = \sum_{i=1}^{q} w_i \mathbf{D}_i \mathbf{n}_i \mathbf{n}_i^t \mathbf{D}_i^t.$$
- The vector $\mathbf{g_N}$ of size l:
$$\mathbf{g_N} = \sum_{i=1}^{q} w_i \mathbf{D}_i \mathbf{n}_i.$$

Therefore, the weighted GOF optimization problem can be rewritten as:

$$min_{\mathbf{a}}\{\mathbf{a}^t(\mathbf{S} + \mu\mathbf{S_N})\mathbf{a} - 2\mu\mathbf{a}^t\mathbf{g_N} + \mu q\}$$

And its solution is obtained solving the following system of linear equations:

$$(\mathbf{S} + \mu\mathbf{S_N})\mathbf{a} = \mu\mathbf{g_N}.$$

Ridge Regression. When the matrix $\bar{\mathbf{S}} = \mathbf{S} + \mu\mathbf{S_N}$ doesn't have a maximal rank or is ill conditioned then the technique called *ridge regression* (RR) can be used. Statisticians use it frequently to remove the collinearity of the input data. The fi rst proposal to obtain algebraic surfaces that fi ts better was done by Tasdizen et al. in [17]. The RR technique basically modifi es the optimization problem of the *gradient one fitting* method by adding a new term:

$$min_{\mathbf{a}}\{\mathbf{a}^t\bar{\mathbf{S}}\mathbf{a} - 2\mu\mathbf{a}^t\mathbf{g_N} + \mu q + \kappa\mathbf{a}^t\mathbf{\Delta}\mathbf{a}\},$$

where $\mathbf{\Delta}$ is a diagonal matrix of size $l \times l$ and the real constant κ determines the weight given to the new term. The minimization problem solution is obtained by solving the following system of linear equations:

$$(\bar{\mathbf{S}} + \kappa\mathbf{\Delta})\mathbf{a} = \mu\mathbf{g_N}$$

2.3 Partition of unity

A *partition of unity* (PU) [7, 11] is a mathematical tool very useful to combine local approximation in order to defi ne a global one. Important properties such as the global maximal error and the convergence order could be inherited from the local approximation. The basic ideas behind the global approximation construction using partition of unity are the following:

1. divide the domain in parts,
2. obtain a local approximation for each part using a subset of the data that belongs to it,
3. obtain a global approximation by the use of a weighted combination of local solution through the use of smooth non-negative functions that correspond to the weights. In each point of the domain, the sum of these weight functions should be one.

More precisely, consider a compact domain $\Omega \subset \mathbb{R}^3$ and denote by $\{\varphi_i\}$ the set of non-negative functions with compact support such that:

$$\sum_i \varphi_i(x, y, z) \equiv 1 \text{ for all points } (x, y, z) \in \Omega.$$

Let $\mathcal{F}_i$ be a set of functions defi ned in $supp(\varphi_i)$. Each function in $\mathcal{F}_i$ represents a local approximation for the points of $\mathcal{P}$ that belong to $supp(\varphi_i)$. A global approximation for the function $f : \Omega \to \mathbb{R}$ could be obtained as following:

$$f(x, y, z) \approx \sum \varphi_i(x, y, z) f_i(x, y, z). \tag{4}$$

where $f_i \in \mathcal{F}_i$. Consider $\{w_i\}$ a set of non-negative functions with compact support such that:

$$\Omega \subset \bigcup_i supp(w_i).$$

The partition of unity functions φ_i could be generated by the following equation:

$$\varphi_i(x, y, z) = \frac{w_i(x, y, z)}{\sum_{j=1}^{n} w_j(x, y, z)} \tag{5}$$

The main idea of partition of unity could be resumed by the equations (8) e (5). Such equations form the basis of the algorithm Multilevel Partition of Unity Implicit (MPU) proposed by Ohtake et al. in [15].

2.4 Multilevel Partition of Unity

The method called *Multilevel Partition of Unity* (MPU) was proposed by Ohtake et al. in [15] originally to build an implicit surface approximation of a set of points and normals in $\mathbb{R}^3$. The MPU uses a partition of unity to obtain a global implicit surface approximation for the boundary of the object combining local approximations. It uses an *Oct-Tree* as an hierarchical scheme to guide the domain subdivision.

Follows a concise description of how the MPU builds an implicit function that globally approximates the points.

The method initially centers the point of $\mathcal{P}$ at the origin. After that, the points are scaled in such a way that the square $\Xi = [-1, 1]^3$ contains all points of $\mathcal{P}$. We will adopt the same name $\mathcal{P}$ for the set of points after these two transformations.

The method builds an *Oct-Tree* by the use of recursive procedure where the subdivision of its node is controlled by the error of the local approximation. In other words, the refi nement criteria for a node i of the *Oct-Tree* consists of computing the local error of the approximation and when this error is greater than a given tolerance, then the node is subdivided in eight new nodes and recursively the same test is again used for each one of its child nodes.

Each node i on the *Oct-Tree* is associated to a weight function w_i with a compact support that is used for the partition of unity global approximation. The compact support of w_i is defi ned as a circle of radius r_i centered on the

middle of the node i. Such radius is chosen proportionally to the size of the diagonal of the square corresponding to the node i, denoted by d_i.

In the MPU original method, a quadratic spline function $b : \mathbb{R} \rightarrow \mathbb{R}$ is used to build the weight function w_i:

$$w_i(x, y, z) = b(\frac{3 \mid (x, y, z) - c_i \mid}{2r_i}) \tag{6}$$

where c_i is the center of the corresponding node i in the *Oct-Tree*. The value of w_i is zero outside the support region. According to the equation (5), the partition of unity function φ_i associated to the node i is defi ned as:

$$\varphi_i(x, y, z) = \frac{w_i(x, y, z)}{\sum_{j=1}^{n_l} w_j(x, y, z)}, \tag{7}$$

where n_l is the number of leaves on the *Oct-Tree*.

The MPU method uses of a quadric (degree 2 polynomial function) to locally approximate the signed distance function to the boundary of the object. To fi nd the coeffi cients of the quadric function $Q : supp(\varphi_i) \subset \mathbb{R}^2 \rightarrow \mathbb{R}$ for the node i, Ohtake et al. used a least squares scheme whose objective function is a weighted mean of the algebraic distance from each point to the boundary, where these weights are computed from w_i itself.

At each node, a least squares problem is solved considering only the points on $\mathcal{P}$ that belong to its support region as input. Sometimes (especially when the density of $\mathcal{P}$ is not uniform) the circle of radius r_i of a node i doesn't contain a suffi cient number of points to estimate robustly the quadric that approximates such points. If the number of points on the support region of a node is not suffi cient to solve the minimization problem, then they adopted a solution that increases the radius of the support region until such minimal condition is guaranteed.

They suppose that the surface $\mathcal{S}$ from where the points are sampled is a level 0 isosurface of a function $f : \mathbb{R}^3 \rightarrow \mathbb{R}$. Then, they can use the partition of unity equation (8) to obtain a $F : \mathbb{R}^3 \rightarrow \mathbb{R}$ that globally approximates f:

$$f(x, y, z) \approx F(x, y, z) = \sum_{j=1}^{n_l} \varphi_i(x, y, z) Q_i(x, y, z). \tag{8}$$

For more information of this method, see [15].

3 The new surface reconstruction scheme

We propose in this section a new method for implicit surface fi tting. It combines the MPU scheme and the *gradiend fitting one* and *ridge regression* methods to improve the implicit surface approximation.

Data input. Similarly to the *gradient one fitting* and to the *ridge regression* methods, our algorithm considers the following data as input:

- a set of q points $\mathcal{P}$, and
- a set of the corresponding q unitary normal vectors $\mathcal{N}$.

We are assuming that the points of $\mathcal{P}$ have been translated in such a way that the center of mass is the origin of the coordinate system, and also that they have been scaled in such a way that all points are contained in the square $\Xi = [-1, 1]^3$. Such square Ξ is the starting region for the hierarchical adaptive space subdivision guided by the use of an *Oct-Tree* data structure.

Support regions. We use the same *Oct-Tree* support region scheme of the MPU method. The support region for each node i is the disk of radius $r_i = \alpha d_i$ centered at c_i.

Local approximations. In our method, we adopt the *gradiend one fitting* and *ridge regression* techniques to obtain the coeffi cients of a degree d algebraic function P_{di} for the local approximation at the node i. The local approximating is only compute when the node contains samples points on its support region. To do this task, we consider the set of points $\mathcal{P}_i = \{p_{j_1}, ..., p_{j_{q_i}}\} \subset \mathcal{P}$ that are on the support region of the node i, in case that we have enough points (a number bigger than $N_{min} = 15$), otherwise we grow the support region until we get a minimum number of points. To run the RR method, we have also to equip each point of $\mathcal{P}_i$ with its unitary normal vectors. Thus, using the same indexes of these vectors we construct the sets $\mathcal{N}_i \subset \mathcal{N}$ for each node i. The objective function to be minimized on each node i is:

$$\{\mathbf{a}^t(\mathbf{S_i}\mathbf{a} + \mu(\mathbf{S_{N,i}})\mathbf{a} - 2\mu\mathbf{a}^t\mathbf{g_{N,i}} + \mu q_i + \kappa\mathbf{a}^t\mathbf{\Delta_i}\mathbf{a}\},$$

where the matrices $\mathbf{S_i}, \mathbf{S_{N,i}}, \mathbf{\Delta_i}$ and the vector $\mathbf{g_{N,i}}$ are computed for the node i using the expressions presented in section 2.2. However, we consider as input for this computation the sets $\mathcal{P}_i$ and $\mathcal{N}_i$. The solution to the minimization problem is obtained by solving the following system of linear equations:

$$(\mathbf{S_i} + \mu\mathbf{S_{N,i}} + \kappa\mathbf{\Delta_i})\mathbf{a} = \mu\mathbf{g_{N,i}}.$$

Oct-Tree construction. The *Oct-Tree* is built using a recursive procedure, whose refi nement criteria is the local approximation error. Consider a tolerance ϵ for the local error. The condition that determines whether node i of the *Oct-Tree* at level l_i should be refi ned is the following boolean expression:

$$((e_i \geq \varepsilon) \text{ and } (l_i < l_{max}) \text{ and } (q_i > N_{min}),$$

where, the local approximation error e_i at node i is the widely known Taubin error metric and also the mean squared algebraic distances from the points $\mathcal{P}_i$ to the obtained surface has been considered. It seems that the Taubin error metric produce good result is less time. In this situation that the *Oct-Tree* node doesn't has points, we copy the coeffi cient of its parent node and stop for this node the refi nement procedure.

Parameters of the method. In conclution, the parameters of the method are the following:

- $d \in \mathbb{N}$: degree of the algebraic surface.
- $l_{max} \in \mathbb{N}$: maximum level for the *Oct-Tree*
- $\alpha \in \mathbb{R}$: constant that multiplies the diagonal size of the node to obtain the radius of the support disk.
- $\mu \in \mathbb{R}$: weight given to the normals on the objective function.
- $\kappa \in \mathbb{R}$: : weight given to the ridge regression additional term on the objective function.
- $\epsilon \in \mathbb{R}$: threshold value for the refi nement condition controlled by the local approximation error.

Notice that with this set of parameters we can unify several methods presented in this paper. For example: if we want to run the original GOF method we have to assign $l_{max} = 0$ and $\kappa = 0$; if we want the original RR method we have only to assign $l_{max} = 0$.

Global approximation and function evaluation. We have supposed that the surface $\mathcal{S} = f^{-1}(0)$ for some function $f : \mathbb{R}^3 \to \mathbb{R}$. Thus, by the use of the partition of unity equation (8) we can obtain a function $F : \mathbb{R}^3 \to \mathbb{R}$ that globally approximates f:

$$f(x, y, z) \approx F(x, y, z) = \sum_{j=1}^{n_l} \varphi_i(x, y) P_{di}(x, y).$$

4 Results

Our method has been tested with several sparse data sets, fi gure5 shows the reconstruction of a sparse set of points sampled from the bunny surface using our methods. Notice that there are some big regions (on the top of the head and the back) without any samples, the algorithm creates some bump at the back which is expected due to the very bad sampling. Also this model has been produce without the ridge regression, that is, $k = 0$.

The fi gure 5 shows the squirrel model which is a point cloud pretty similar to the ones which are produce from a 3D scanner. This data set had missed the bottom, that is, there is a big hole at the bottom. Our method with $k = 0.0001$ was able to reconstruct the model without creating any spurious connected component that could appeared due to a bad fi tting near the bottom of the squirrel model.

Without the use of ridge regression the algorithm was unable to reconstruct the knot model, see central picture of fi gure 5, in which some small components appear, but the problem can be fi xed using ridge regression with parameter $k = 0.0001$, see right picture of fi gure5.

Finally the fi gure5 shows the algorithm using the Gradient one fi tting method without weights

$$e_{grad} = \sum_{i=1}^{q} (P_d(x_i, y_i, z_i)^2 + \mu(\mathbf{n}_i^t \nabla P_d(x_i, y_i, z_i) - 1)^2 \tag{9}$$

from left to right different values of the ridge regression parameter k have been tested ($k = 0$, $k = 0.001$, $k = 0.0001$ and $k = 0.00001$) for each of those values of k the algorithm seems to created the same undesired connected components. This confi rm the role of the weighed Gradient one fi tting together with the ridge regression technique to give some regularization to this ill pose problem.

5 Conclusions and future works

We proposed a new method that combines two powerful techniques: the *weighted gradient one fitting + ridge regression* and the *multilevel partition of unity*. On one side, the *ridge regression* method has been considered by the pattern analysis community as one that gives a better fi tting, since it tries to have a correct topology on the surface reconstruction. However, when the surface has a complex shape it is necessary to elevate the degree of the algebraic surface to get a good result. On the other side, the *multilevel partition of unity* is an implicit method that is now one of the most important reconstruction techniques. In order to compute local approximations, it uses a complicate objective function. Thus, our surface reconstruction scheme not only takes the advantage of these two well recognized methods, but also unifi es those methods in a simple setting.

We plan to continue this work in several directions: one possible direction is to determine an approximated tangent plane T to the samples on the node of the tree and over this plane we consider the surface as a hight fi eld and determine a bivariate local approximation $Q(x, y) : T \to \mathbb{R}$. Also over T we can consider better approximations like for example using wavelets method in order to be able to faithfully reproduce the oscillations and details (texture) on each region. Other promising approach is to combine this powerful method with interval arithmetic meshing algorithm in order to get an accurate and robust representation of the geometry and topology of the 3D object.

References

[1] M. Alexa, J. Behr, D. Cohen-Or, S. Fleishman, D. Levin, and C. T. Silva. Point set surfaces. In *IEEE Visualization 2001*, pages 21–28, October 2001.

[2] Marc Alexa, Johannes Behr, Daniel Cohen-Or, Shachar Fleishman, David Levin, and Claudio T. Silva. Computing and rendering point set surfaces. *IEEE Trans. on Visualization and Computer Graphics*, 9(1):3–15, 2003.

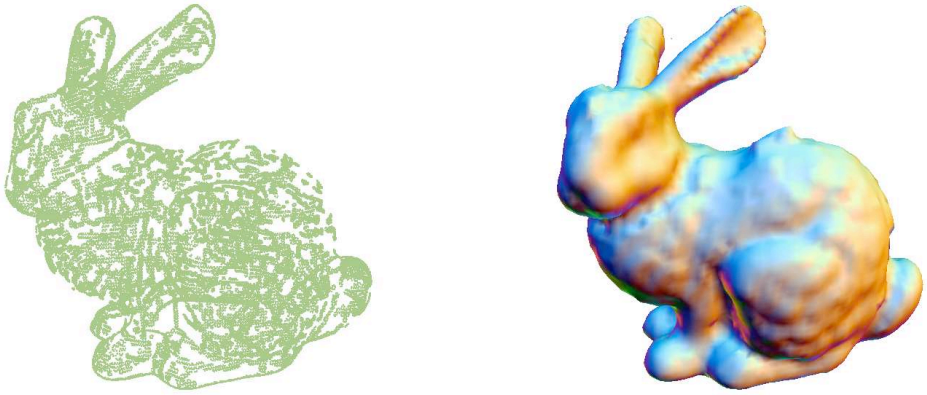

Figure 1: Bunny model. Left, a sparse point cloud. Right, the reconstructed model

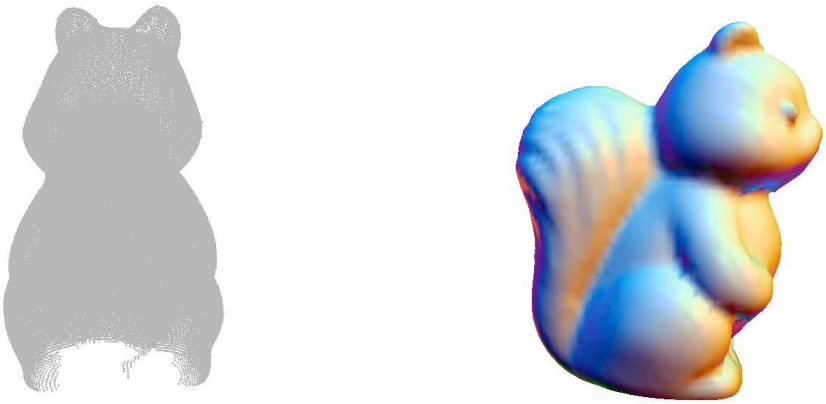

Figure 2: Squirrell model. Left, a point cloud with some holes, specially the one at the buttom. Right, the reconstructed model.

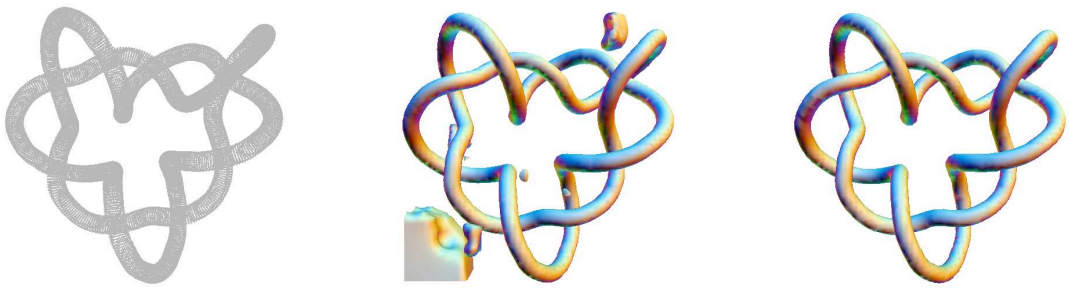

Figure 3: Knot model. Left, point cloud. Center, reconstructed model with ridge regression parameter $k = 0$, some undesired components appears. Right, a perfect reconstruction model with $k = 0.0001$, the spurius components desapeared

Figure 4: Knot model reconstructed using gradient one fi tting technique without the use of weights. Different values of k (from left to right $k = 0$, $k = 0.001$, $k = 0.0001$ and $k = 0.00001$) have been considered and not improvement is observed

[3] N. Amenta and M. Bern. Surface reconstruction by voronoi fi ltering. *Discrete and Computational Geometry*, 22(4):481–504, 1999.

[4] N. Amenta, S. Choi, T. K. Dey, and N. Leekha. A simple algorithm for homeomorphic surface reconstruction. *Symposium on Computational Geometry*, pages 213–222, 2000.

[5] Nina Amenta, Sunghee Choi, and Ravi Krishna Kolluri. The power crust, unions of balls, and the medial axis transform. *Computational Geometry*, 19(2-3):127–153, 2001.

[6] S. Arouca. M´etodo impl´ı cito para reconstrucao de curvas a partir de pontos esparsos. *Master degree thesis at PUC*, 2005.

[7] I. Babuska and J. M. Melenk. The partition of unity method. *International Journal of Numerical Methods in Engineering*, (40):727–758, 1997.

[8] Michael M. Blane, Zhibi Lei, Hakan Çivi, and David B. Cooper. The 3l algorithm for fi tting implicit polynomial curves and surfaces to data. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(3):298–313, 2000.

[9] J. C Carr, R. K. Beaston, J. B. Cherrie, J. B. Michell, T. J Friaght, W. R McCallum, and T. R. Evans. Reconstruction and representation of 3d object with radial bases functions. In *Proceedings of the ACM SIGGRAPH 2001*, pages 67–76, 2001.

[10] T Dey and S. Goswamy. Provable surface reconstruction from noisy samples. *Comput. Geom*, 35(1-2):124–141, 2006.

[11] R. Franke and G. Nielson. Smooth interpolation of large sets of scattered data. *International Journal of Numerical Methods in Engineering*, (15):1691–1704, 1980.

[12] David Levin. Mesh-independent surface interpolation. *Geometric Modeling for Scientific Visualization*, to appear, 2003.

[13] Marc Levoy, Kari Pulli, Brian Curless, Szymon Rusinkiewicz, David Koller, Lucas Pereira, Matt Ginzton, Sean Anderson, James Davis, Jeremy Ginsberg, Jonathan Shade, and Duane Fulk. The digital michelangelo project: 3D scanning of large statues. In *ACM Siggraph 2000*, pages 131–144, 2000.

[14] B. Mederos, N. Amenta, L. Velho, and L de Fuiguereido. Surface reconstruction for noisy point clouds. In *Symposium on Geometry Processing*, pages 53–62, Viena, Austria, 2005.

[15] Yutaka Ohtake, Alexander Belyaev, Marc Alexa, Greg Turk, and Hans-Peter Seidel. Multi-level partition of unity implicits. *ACM Trans. Graph.*, 22(3):463–470, 2003.

[16] V. V. Savchenko, A. A. Pasko, O. G. Okunev, and T. L. Kuni. Function reprresntation of solid reconstructed from scattered surface points and contours. *Computer Graphics Forum*, (14):181–188, 2002.

[17] T. Tasdizen, J. P. Tarel, and D. B. Cooper. Algebraic curves that work better. In *IEEE Conf. on Comp. Vision and Pattern Rec.*, volume II, pages 35–41, 1999.

[18] Gabriel Taubin. Estimation of planar curves, surfaces, and nonplanar space curves defi ned by implicit equations with applications to edge and range image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 13(11):1115–1138, 1991.

[19] G. Turk and O'brian. Modeling with implicit surfaces that interpolates. *ACM transaction on Graphics*, (21):855–873, 2002.