SIMS: a Multi-Level Approach to Surface Reconstruction with Sparse Implicits

Giuseppe Patané
IMATI-CNR, Genova - Italy
patane@ge.imati.cnr.it

Abstract

3D shape approximation and processing with implicitly defined surface primitives (Radial Basis Functions and more general kernel-based approximations, Partition of Unity approximations, Moving Least Squares, etc.) are currently a subject of intensive research in Geometric Modeling and Computer Graphics. In this paper, we propose an approach that combines two conflicting criteria: achieving a high approximation accuracy and obtaining an economical surface representation. We employ compactly-supported Radial Basis Functions and use Tikhonov Regularization to achieve a near optimal selection of their centers. An iterative approach, which defines a multi-level approximation, is used to cope with arising constrained optimization problems.

1 Introduction

Current scanning techniques are capable of acquiring large and redundant data sets, thus arising the problem of defining approximations that use only a small percentage of the available information. On discrete 3D surfaces, a common approach to approximating, reducing, and hierarchically organizing the input data is provided by simplification and multi-resolution techniques [2, 14, 20].

Implicit modeling [7] represents a valid and equally expressive alternative to the above-mentioned techniques. For instance, implicit reconstruction does not impose constraints on the topological complexity of the input data and guarantees that the reconstructed surface is water-tight. In this case, a 3D data set $\mathcal{P}$ is approximated by an implicit surface $\Sigma := \{p \in \mathbb{R}^3 : f^{\star}(p) = 0\}$, and the function $f^{\star}(x) = \sum_{i=1}^{N} a_i\varphi_i(x)$ is a linear combination of the basis elements $\mathcal{B} := \{\varphi_i(x)\}_{i=1,\dots,N}$. The underlining mathematical framework builds on numerical linear algebra and the degrees of freedom on the choice of $\mathcal{B}$ (*e.g.*, globally [9, 39] and compactly supported RBFs [24, 26, 28], Partition of Unity [25, 41], Moving-Least-Squares methods [1, 17, 22, 23, 30, 34]) enable to adapt the model parameters to specific problem constraints such as huge data sets with attributes, local accuracy, and degree of smoothness. Furthermore, multi-resolution techniques have been recently proposed [37] and a wide range of applications, which include deformation, fast rendering, and collision detection [3, 27, 38], have been targeted by several authors during the last decades.

In literature [11], the use of linearly dependent basis functions (*i.e., overcomplete*) arised the problem of selecting among several representations of $f^{\star}$ the one which involves the smallest number of basis functions. Since we use linearly independent basis functions, we consider all the sub-bases $\mathcal{B}^{'} := \{\varphi_i(x) : i \in I^{'} \subseteq I\}$ of $\mathcal{B}$, $I := \{1, \dots, N\}$, and we reformulate the problem as searching the approximation $\hat{f}(x) = \sum_{i \in I'} a_i\varphi_i(x)$, which is the best compromise among the following criteria:

1. *sparsity*: we should find a representation $\hat{f}$, called *sparse approximation*, with the smallest number of significant coefficients. This aim is usually achieved by selecting the function $f^{\star}$ with the smallest l_1-norm of coefficients among all such decompositions. Instead of using the l_1-norm, which is not differentiable, we introduce a C^1-approximation and study its sparsification and regularization properties;

2. *approximation accuracy*: we should obtain the highest accuracy among all possible decompositions;

3. *smoothness and feasible computational cost*: the above-mentioned criteria must guarantee a smooth solution and $O(N \log N)$ computational cost.

The problem of building sparse approximations has a theoretical importance and usefulness for interactive shape modeling and processing, where a minor number of basis functions speed up the surface sampling, interactive modeling and queries [8, 35].

While previous work [33, 35] has been focused on Support-Vector Machine (SVM) [13, 31], in this paper we consider an equivalent formulation [18] provided by Tikhonov regularization theory [6, 36]. In this way, we introduce a novel approach to sparse approximation that replaces the large and constrained minimization problem related to SVM with an approximate formulation, which is defined by a system of non-linear equations. This choice enables to substitute heuristic and decomposition methods for constrained convex minimization problems with standard iterative solvers of non-linear equations. Therefore, we provide a *multi-level approach to Sparse IMplicit approximations* (SIMS for short) and we prove that

- it achieves a sparse approximation while preserving properties of the input kernel and that correspond to sparsity and positive definiteness of its Gram matrix;

- it generates a *multi-level approximation scheme* represented by a sequence of *nested* spaces $(V_n)_n$, where each sparse approximation $f^{(n)}$ of $f^\star$ belongs to V_n. This property is a new feature provided by SIMS and not shared by previous work.

We note that multi-resolution techniques such as [37] are not defined on the implicit representation of $f^\star$ but on the sampling and polygonalization process; therefore, the resolution of the voxel grid determines the complexity of the reconstructed surface. In [27], a multi-scale representation of a smooth surface is achieved by applying a local polynomial approximation, whose number of centers and radii are adapted to the required resolution. This local approximation makes the approach best-suited for interactive visualization and feature extraction of large-size models.

The main difference with respect to our approach is that SIMS achieves a compact representation through a global procedure, which builds on well-defined properties such as approximation accuracy and smoothness. Therefore, it avoids a-posteriori updates of the model that are based on the evaluation of the reconstruction error. Since we use a global formulation of the sparsification problem, the center selection is achieved by a *top-down* procedure instead of a *bottom-up* structure as done by local methods [9, 16, 28, 29]. However, as discussed in Section 3.3, bottom-up approaches to surface reconstruction (*e.g.*, [9]) can use SIMS to sparsify the implicit approximation.

The paper is organized as follows: Section 2 describes previous work on implicit and sparse approximation based on Regularization Theory and Support Vector Machine. Section 3 defines our approach and its main features. Section 4 discusses numerical aspects and results; future work is presented in Section 5.

2 Regularization and Support Vector Machine

Let $\mathcal{P} := \{(x_i, y_i)\}_{i=1}^N$ be an input data set, possibly affected by noise, and associated to an unknown function $f : \mathbb{R}^d \rightarrow \mathbb{R}$ such that $f(x_i) = y_i, i = 1, \dots, N$. The function f is approximated by assuming that it belongs to a linear space $\mathcal{H}$ spanned by a set of linearly independent basis functions $\{\varphi_i : i = 1, \dots, +\infty\}$, that is,

$$f(x) = \sum_{i=1}^{+\infty} \alpha_i \varphi_i(x). \tag{1}$$

Since (1) is ill-posed, Tikhnonov [36] proposed to approximate f with the solution of the following minimization problem

$$\min_{f \in \mathcal{H}} \left\{ \alpha \sum_{i=1}^N \Gamma(y_i - f(x_i)) + \beta \Phi(f) \right\}, \tag{2}$$

where α and β are positive constants that define the trade-off between approximation error, measured by the cost function Γ, and smoothness of the solution provided by the regularization functional Φ defined on $\mathcal{H}$ (*e.g.,* differential operators). Under mild conditions on Φ [6, 36], the problem (2) is well-posed. Common choices are:

- *Least-squares regularization* [6]

  $$\Gamma(t) := t^2, \quad \beta := (1 - \alpha), \quad 0 \leq \alpha \leq 1;$$

- *Support Vector Machine* [13]

  $$\Gamma(t) := |t|_\epsilon := \begin{cases} 0 & \text{if } |t| < \epsilon, \\ |t| - \epsilon & \text{otherwise} \end{cases}$$

  where $| \cdot |_\epsilon$ is called *ϵ-insensitive cost function.*

Previous work on data approximation involves the use of the Fourier basis, orthonormal wavelet bases, and discrete cosine transform bases [10, 11].

Following the approach proposed in [18], we can choose as $\mathcal{H}$ a *Reproducing Kernel Hilbert Space* (RKHS) with kernel function $K(x, y)$ (*e.g.*, the Gaussian kernel $K(x, y) := \exp(-\|x - y\|^2)$ or the polynomial of degree d $K(x, y) := (1 - xy)^d$) and basis

$$\varphi_i(x) := K(x, x_i), \quad i = 1, \dots, N.$$

Among the properties of RKHS, we remind the *reproduction property*

$$h(x) =< h(y), K(x, y) >_{\mathcal{H}}, \qquad \forall h \in \mathcal{H}, \qquad \forall x, y \in \mathbb{R}^d, \tag{3}$$

that will be used in Section 3 to derive a sparse approximation of f; for a detailed discussion on RKHS, we refer the reader to [4].

Let

$$f^{\star}(x) := \sum_{i=1}^{N} a_i K(x, x_i) \tag{4}$$

be an approximation of f, $\frac{1}{2}\|f - \sum_{i=1}^{N} a_i K(x,x_i)\|_{\mathcal{H}}^2$ the quadratic misfit measure on the input data, and $\|a\|_{l_1} := \sum_{i=1}^{N} |a_i|$ the l_1-norm of the coefficients $a := (a_i)_{i=1}^{N}$ occurring in the representation (4). Then, the function which is the best compromise between approximation accuracy and sparse representation is the solution of the following problem

$$\min_{f^{\star}\in\mathcal{H}} \left\{ \frac{1}{2}\|f - \sum_{i=1}^{N} a_i K(x,x_i)\|_{\mathcal{H}}^2 + \epsilon\|a\|_{l_1} \right\} \tag{5}$$

where ϵ, $\epsilon > 0$, is the tradeoff between the misfit measure and sparsity.

Comparing (2) with (5), we note that the regularization component $\Phi(f)$ has been included in the first term of (5), while the term $\|a\|_{l_1}$ aims at minimizing the number of non-zero coefficients in (4). It follows that the smoothness of the solution guarantees the robustness of $f^{\star}$ to noise, and its sparse representation enables to select from a large set of basis functions only those that are strictly necessary to achieve the required trade-off.

In (5), replacing each unknown a_i with a pair of positive variables (a_i^+, a_i^-), such that $a_i = a_i^+ - a_i^-$, brings to the following constrained convex quadratic optimization problem (*i.e.*, Support Vector Machine) [18]:

$$\min_{a_i^+, a_i^- \in \mathcal{S}} \left\{ F(a^+, a^-) \right\} \tag{6}$$

with

$$F(a^+, a^-) : = \frac{1}{2}\sum_{i,j=1}^{N} K(x_i, x_j)(a_i^+ - a_i^-)(a_j^+ - a_j^-) - \sum_{i=1}^{N} y_i(a_i^+ - a_i^-) + \epsilon\sum_{i=1}^{N}(a_i^+ + a_i^-),$$

and feasible set

$$\mathcal{S} := \{a^+ \geq 0,\ a^- \geq 0,\ \sum_{i=1}^{N}(a_i^+ - a_i^-) = 0, \\ a_i^+ a_i^- = 0,\ i = 1, \dots, N\}.$$

This last formulation is equivalent to (5) and facilitates its numerical optimization by removing the absolute values. In [33, 35], (6) is solved by applying a decomposition method and a heuristic coordinate descendent optimization scheme respectively. In the last case, at each iteration $(2N - 1)$ variables are fixed and the objective function is minimized with respect to the remaining free parameter. If this solution does not belong to the interval $[0, \epsilon]$, it is set to its closest feasible point.

3 "Iterative" sparse approximation

If $\mathcal{P}$ is a large data set, defining a sparse representation of f as solution of the quadratic minimization problem (6) with $2N$ unknowns is generally unfeasible due to the amount of input data, the possible ill-conditioning of the coefficient matrix $K := (K_{ij})_{i,j=1,\dots,N}$, $K_{ij} := K(x_i, x_j)$, and the unbounded feasible set $\mathcal{S}$. Therefore, all these factors badly affect the stability and convergence of iterative methods [12, 19]. To overcome the above-mentioned drawbacks and define a multi-level sparse approximation (see Section 3.1), SIMS replaces (5) with an approximated formulation, whose solution does not require to solve a constrained quadratic minimization problem but a system of non-linear equations.

To this end, we introduce the functional

$$\mathcal{F}[f^{\star}] := \frac{1}{2}\|f - \sum_{i=1}^{N} a_i K(x,x_i)\|_{\mathcal{H}}^2 + \epsilon\sum_{i=1}^{N}[a_i^2 + \eta]^{1/2} \tag{7}$$

where η is a positive parameter, and the term $\sum_{i=1}^{N}[a_i^2 + \eta]^{1/2}$ is a smooth approximation of $\|a\|_{l_1}$ for $\eta \to 0^+$. This approximation of the l_1-norm was used in [15] for image processing.

Using the reproduction property (3), we can rewrite (7) as

$$\mathcal{F}[f^{\star}] \equiv \frac{1}{2}\|f\|_{\mathcal{H}}^2 - \sum_{i=1}^{N} y_i a_i + \frac{1}{2}\sum_{i,j=1}^{N} K(x_i, x_j) a_i a_j + \epsilon\sum_{i=1}^{N}[a_i^2 + \eta]^{1/2},$$

where the term $\frac{1}{2}\|f\|_{\mathcal{H}}^2$ is constant. Then, the critical points of $\mathcal{F}$ are achieved as solution of the following system of non-linear equations

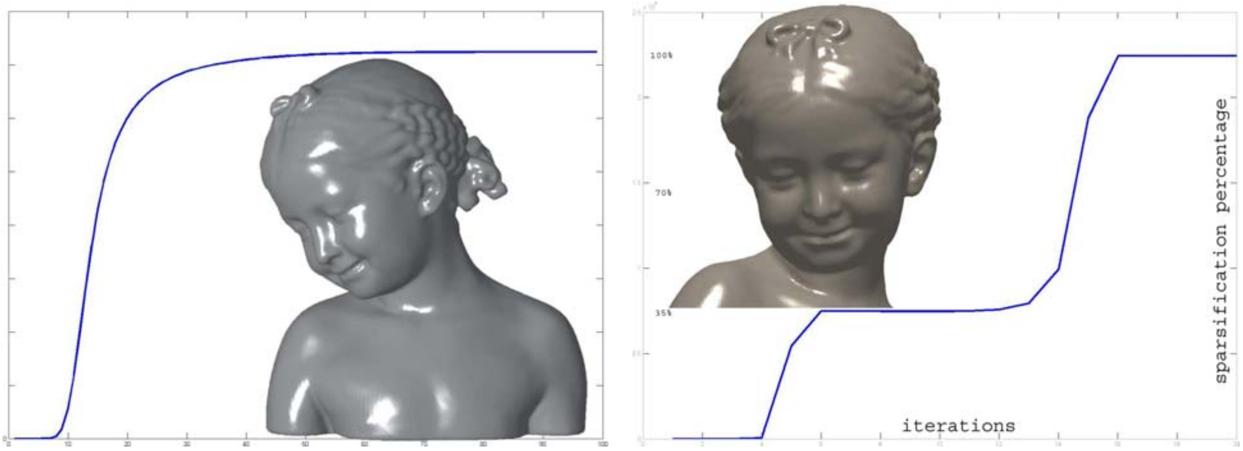
$$\frac{\partial \mathcal{F}}{\partial a_j} = 0 \leftrightarrow \sum_{i=1}^{N} K(x_i, x_j) a_i + \epsilon\frac{a_j}{[a_j^2 + \eta]^{1/2}} = y_j, \tag{8}$$

$j = 1, \dots, N$, that is,

$$[K + \epsilon\Delta(a)]\, a = y \tag{9}$$

with

$$\Delta(a) := \text{diag}\left(\frac{1}{[a_1^2 + \eta]^{1/2}}, \dots, \frac{1}{[a_N^2 + \eta]^{1/2}}\right),$$



(a) (b)

Figure 1. Plot of the (a) partial and (b) full sparsification function Θ with $\epsilon = 0.4$ and $\epsilon = 1.2$ respectively. The surface reconstruction in (a) (resp., (b)) used the 50% (resp., 62%) of input centers ($N = 240K$). On the x-axis, the number of iterations is shown while the corresponding number of null coefficients is given on the y-axis.

and $y := (y_i)_{i=1}^N$. The Hessian matrix $H(\mathcal{F}) := \left(\frac{\partial^2 \mathcal{F}}{\partial a_i \partial a_j}\right)_{ij}$ is given by

$$\begin{cases} K(x_i, x_j) & i \neq j \\ K(x_j, x_j) + \frac{\epsilon\eta}{(a_j^2+\eta)^{3/2}} & \text{else} \end{cases}$$

or, in matrix form,

$$H(\mathcal{F}) = K + \epsilon\eta[\Delta(a)]^3. \qquad (10)$$

If K is positive or semi-positive definite (*e.g.*, Gaussian kernel, compactly supported RBFs), the Hessian matrix (10) is positive definite for all $a \in \mathbb{R}^N$, $\eta > 0$, and $\epsilon > 0$. Therefore, $\mathcal{F}$ is a strictly convex functional whose unique minimum is the solution of equation (9), and it is calculated by applying an iterative scheme

$$\begin{aligned} &\left[K + \epsilon\Delta(a^{(n)})\right] a^{(n+1)} = y \ \leftrightarrow \\ &a^{(n+1)} := \left[K + \epsilon\Delta(a^{(n)})\right]^{-1} y, \quad n \geq 1 \end{aligned} \qquad (11)$$

with $a^{(0)}$ initial guess.

Similar considerations hold if we take into account in (5) the l_0-norm, which is approximated in (7) as

$$\|a\|_{l_0} \approx \sum_{i=1}^N \left[1 - \exp\left(-\frac{a_i^2}{2m^2}\right)\right], \qquad m \to 0^+,$$

and whose diagonal matrix is

$$\Delta(a) := \frac{1}{m^2}\text{diag}\left(\exp\left(-\frac{a_1^2}{2m^2}\right), \ldots, \exp\left(-\frac{a_N^2}{2m^2}\right)\right).$$

In the following of the paper, we refer to (7) and (9) as "*iterative sparse approximation*"; an example of center selection and reconstruction is shown in Figure 1. A detailed discussion on the choice of $a^{(0)}$, the solution of the linear system (11) at each iteration n, and the stop criteria is given in Section 4.

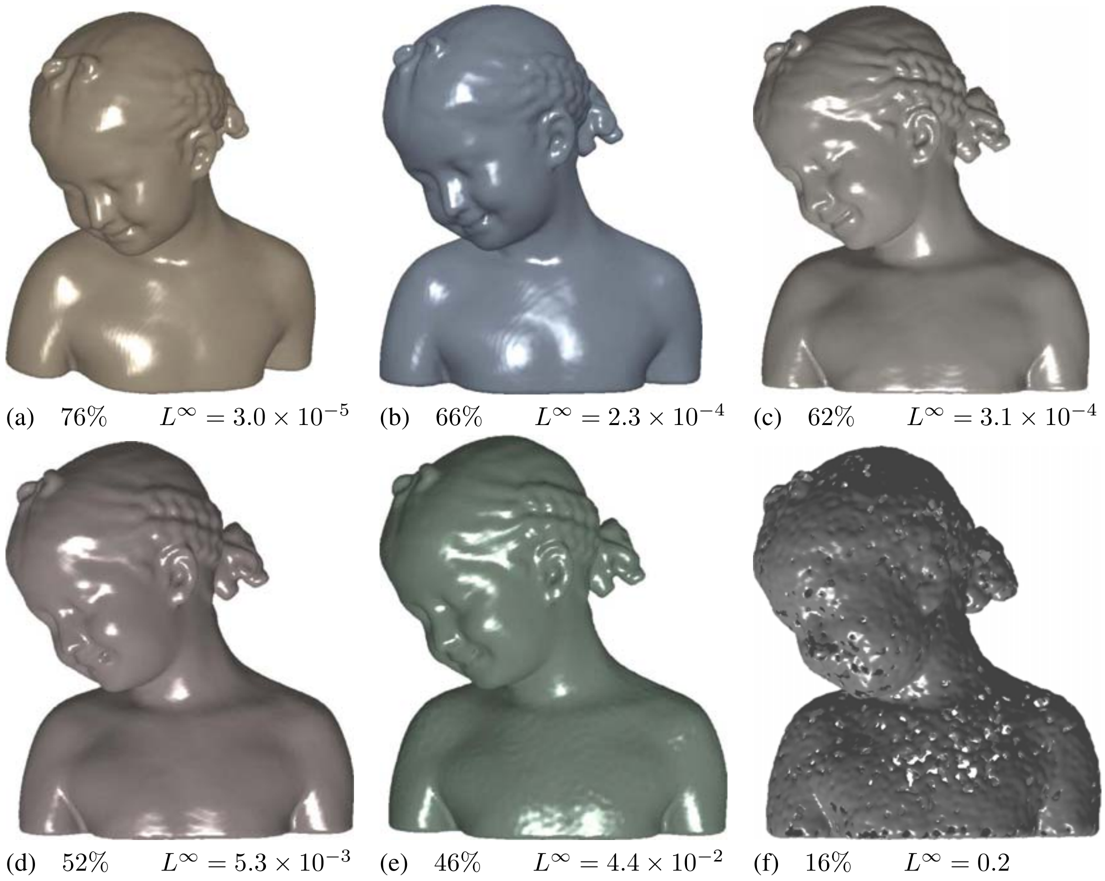
3.1 Multi-level sparse approximation

In this section, we analyze the properties of SIMS and we prove that, fixed the parameter ϵ, (11) induces a *multi-level sparse approximation*. At each iteration $n \geq 1$, we consider the set of indices $I_n := \{i : a_i^{(n)} = 0\}$ related to the basis functions that do not contribute to $f^{(n)}$; then, the sparse approximation $f^{(n)}$ at level n is

$$f^{(n)}(x) := \sum_{i \in I_n^C} a_i^{(n)} \varphi_i(x)$$

where I_n^C is the complement of I_n. From equation (11) it follows that

$$\sum_{j=1}^N K_{ij} a_j^{(n+1)} + \frac{\epsilon}{[a_i^{(n)2} + \eta]^{1/2}} a_i^{(n+1)} = y_i,$$



(a) 76% $L^\infty = 3.0 \times 10^{-5}$ (b) 66% $L^\infty = 2.3 \times 10^{-4}$ (c) 62% $L^\infty = 3.1 \times 10^{-4}$

(d) 52% $L^\infty = 5.3 \times 10^{-3}$ (e) 46% $L^\infty = 4.4 \times 10^{-2}$ (f) 16% $L^\infty = 0.2$

Figure 2. Reconstruction with compactly supported radial basis functions on a model with $240K$ input centers and different percentages of selected centers; the sparsification plot is given in Figure 1(b). In the examples of the paper, we evaluate the approximation error between f and $f^{(n)}$ as $L^\infty := \max_{i=1,\dots,N}\{|f(p_i) - f^{(n)}(p_i)|\}$.

$i = 1, \dots, N$; therefore, for $i \in I_n$ we have

$$\sum_{j=1}^{N} K_{ij} a_j^{(n+1)} + \frac{\epsilon}{\eta^{1/2}} a_i^{(n+1)} = y_i,$$

or equivalently,

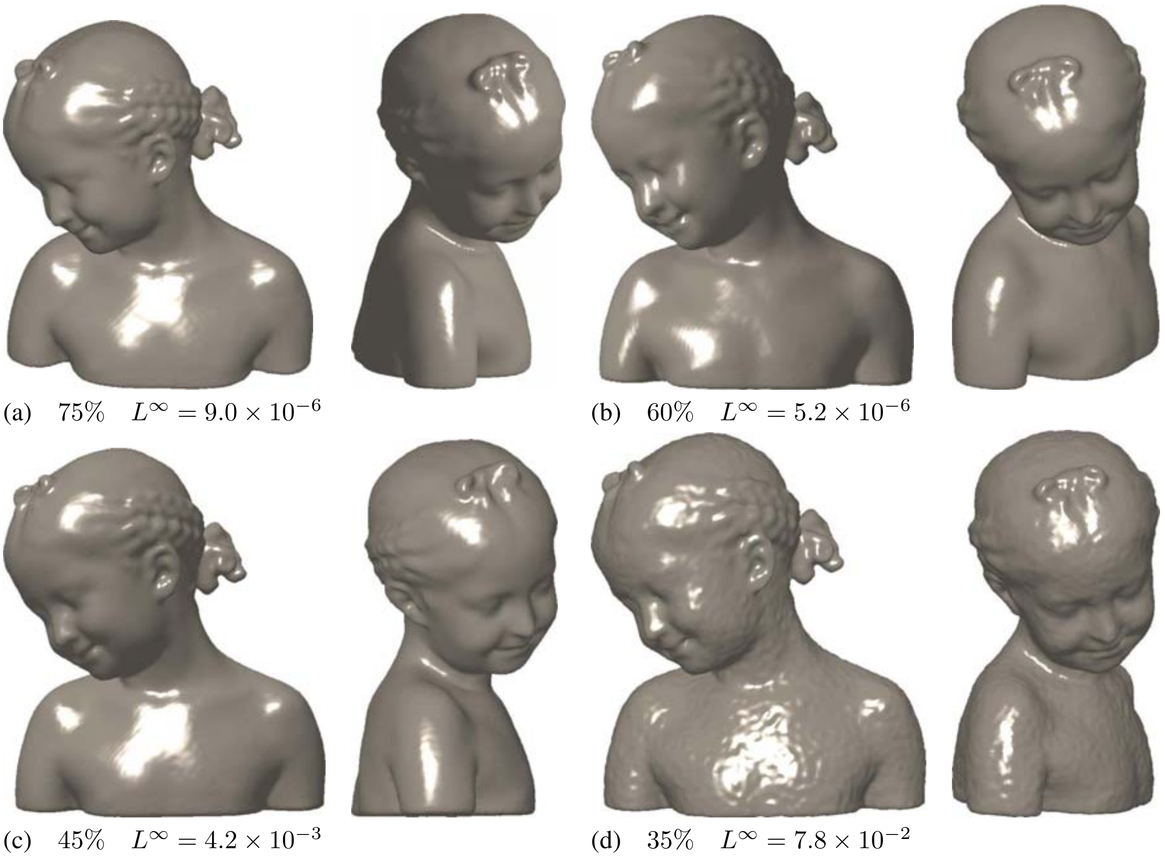
$$a_i^{(n+1)} = \left[K_{ii} + \frac{\epsilon}{\eta^{1/2}}\right]^{-1} \left[y_i - \sum_{j \neq i} K_{ij} a_j^{(n+1)}\right] \text{[1]}. \tag{12}$$

Since $K_{ii} \geq 0$, $\eta << \epsilon$, and $\eta \approx 0$ we obtain $a_i^{(n+1)} \approx 0$, $i \in I_n$, and therefore $I_n \subseteq I_{n+1}$.

We now analyze the behavior of the discrete function $\Theta : \mathbb{N} \rightarrow \mathbb{N}$, $\Theta(n) := \sharp\{i : a_i^{(n)} = 0\}$, which defines the number of zero coefficients in $f^{(n)}$ at level n. From the previous discussion, it follows that Θ is non-decreasing and $\lim_{n \rightarrow +\infty} \Theta(n) = N - \overline{n}$ where $\overline{n}$ is the number of non-zero coefficients of $f^\star$. This last relation ensures that the number of null coefficients $\sharp I_n$ cannot decrease and it defines a multi-level sparse approximation. More precisely, let V_n be the linear space spanned by the basis functions $\{\varphi_i,\ i \in I_n^C\}$, $V := \text{span}\{\varphi_i : i = 1, \dots, N\}$, and $m \geq n$ (*i.e.*, $I_m^C \subseteq I_n^C$); then, $(V_n)_n$ is a sequence of nested spaces $V_m \subseteq V_n$ contained in V such that $f^{(n)} \in V_n$. Therefore, the function Θ sorts the input centers in increasing order of importance with respect to the approximation task.

In the following, we discuss the influence of the kernel K, its support, and the parameter ϵ on the multi-level sparse approximation. Figures 2 and 3 show the reconstruction achieved with compactly supported RBFs of different support and at several levels of detail, *i.e.,* with a different spar-

[1] Since $(a^{(n)})_n$ is convergent, the second term of this equation is bounded.



(a) 75% $L^{\infty} = 9.0 \times 10^{-6}$

(b) 60% $L^{\infty} = 5.2 \times 10^{-6}$

(c) 45% $L^{\infty} = 4.2 \times 10^{-3}$

(d) 35% $L^{\infty} = 7.8 \times 10^{-2}$

Figure 3. In this test, the support of the RBFs used is twice of the one used in Figure 2; comparing the reconstructed surfaces we see how a greater support enables to achieved a better approximation of local details and a higher sparsification rate without holes.

sification percentage. Comparing Figures 2(a-e) with (f), it follows that the removal of the centers can create holes in the reconstructed surface; this phenomenon is due to the use of compactly supported RBFs with a small support. As shown in Figure 3 and 4, the increase of the support removes artifacts, enables to select a greater sparsification rate, but reduces the sparsity of the coefficient matrix K. In Figure 5, globally supported RBFs guarantee the topological coherence of the reconstructed surface until a high sparsification percentage is reached.

Since ϵ controls the compromise between approximation error and sparse solution, a low value of this parameter results in a good reconstruction and a low number of null coefficients. Increasing ϵ forces the iterative method to converge to the null solution (*i.e., complete sparsification*) and generates the whole multi-level scheme (see Figure 1(b)) and smoother results. For instance, in Figure 4(b), the regularization term has dominated the sparsification step and it results in a smooth reconstructed surface.

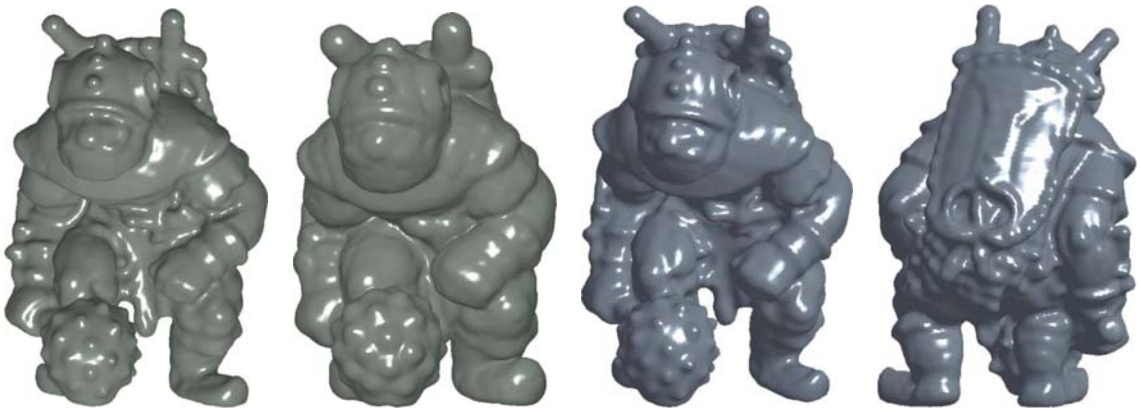
3.2 Iterative least-squares sparse approximation

In the definition of the functional (7), we considered a set of basis functions centered at each point of $\mathcal{P}$. However, when we search a sparse representation of $f^{\star}$ it is not necessary to use all the input points as centers if:

- we assume to deal with highly redundant data, or
- we aim at further reducing the computational cost of the sparsification scheme.

Therefore, another choice consists of replacing (5) with a least-squares formulation

$$\min_{f^{\star} \in \mathcal{H}} \left\{ \sum_{i=1}^{N} |f(x_i) - f^{\star}(x_i)|^2 + \epsilon \|a\|_{l_1} \right\} \qquad (13)$$



(a) 94% $k = 20$ (b) 66% $k = 20$ (c) 52% $k = 40$

Figure 4. Input dataset with 78K centers and reconstructed surfaces with a different number of selected centers; choosing a larger support of the basis functions (*i.e.*, a larger k in the construction of the kd-Tree) guarantees to increase the sparsification rate as well to recover local details.

and $f^\star(x) := \sum_{i=1}^{M} a_i K(x, c_i)$, where $\mathcal{C} := \{c_i\}_{i=1,\ldots,M}$, $N > M$, is the initial set of centers (*e.g.*, $\mathcal{C} = \mathcal{P}$). In this case, we are reducing the degree of smoothness of f provided by $\|\cdot\|_{\mathcal{H}}$ while increasing the number of constraints on its shape. The analysis of the critical points of (13) leads to replace (8-9) with

$$\sum_{i=1}^{M}\sum_{j=1}^{N} K(x_j, c_i)K(x_j, c_k)a_i + \epsilon\frac{a_k}{[a_k^2 + \eta]^{1/2}} = \sum_{j=1}^{N} K(x_j, c_k)y_j, \quad k = 1, \ldots, M$$

and

$$\left[K^T K + \epsilon\Delta(a)\right] a = K^T y, \quad K := (K(x_i, c_j))_{i=1,\ldots,N}^{j=1,\ldots,M} \tag{14}$$

whose dimension is $M \times M$ instead of $N \times N$.

We note that, for each $a \in \mathbb{R}^M$ and $\epsilon > 0$, $(K^T K + \epsilon\Delta(a))$ is positive-definite without assumptions on the matrix K; on the contrary, in Section 3 we assumed to deal with a semi-positive definite matrix. Furthermore, replacing $\|f - f^\star\|_{\mathcal{H}}$ with $\sum_{i=1}^{N} |f(x_i) - f^\star(x_i)|^2$ is necessary if we choose a set of basis functions that does not satisfy the reproduction property. We refer to (13) and the corresponding normal equation (14) as *least-squares sparse approximation* (see Figure 6); finally, we underline that the discussion in Section 3.1 still applies by substituting K with $K^T K$, y with $K^T y$, and N with M.

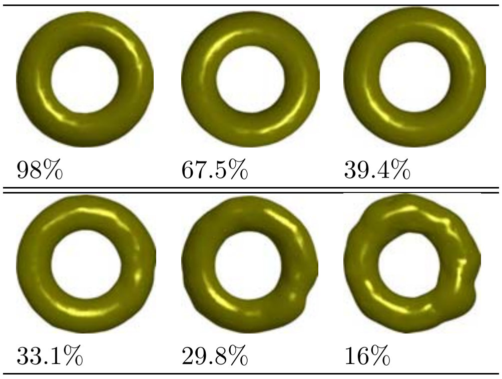


Figure 5. Sparsification and reconstruction of the bitorus; the use of globally supported basis functions, associated to the kernel $K(x, y) := \exp(-\|x - y\|^2)$, guarantees a good reconstruction until the sparsification rate becomes too high (*i.e.*, greater than 60% of the input centers).

3.3 Integrating SIMS with bottom-up approaches to surface approximation

To reduce the computational cost of the above scheme, which can require to deal with a problem of dimension $N \times N$ up to $3N \times 3N$, we can integrate SIMS with *bottom-up* approaches that iteratively updates the number of centers by adding only those that are strictly necessary to improve the approximation accuracy. The basic idea is to support the

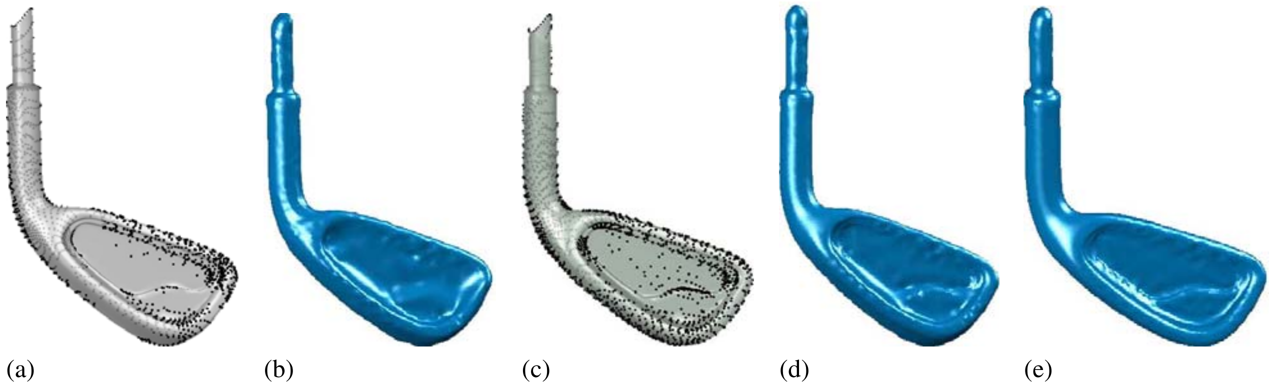


Figure 6. Iterative least-squares sparse approximation with $N = 2407$ input centers placed in $\mathcal{P}$: (a-b) $\epsilon = 0.05$, $\overline{n} = 1050$ selected centers (*i.e.*, 43.63%), (c-d) $\epsilon = 0.01$ and $\overline{n} = 1568$ (*i.e.*, 65.14%), (e) $\epsilon = 0$.

procedure discussed in [9] with a sparsification step, which controls the trade-off between the reduction of the local error and the number of centers.

Starting from a set of centers chosen on the input data set using a clustering technique, let $\overline{f}$ be the current approximation of f with respect to a set of centers $\mathcal{C}$. At the next iteration, we add to $\mathcal{C}$ all those points $\mathcal{Q} \subseteq \mathcal{P}$ where the approximation error is greater than a given threshold δ (*i.e.*, $p \in \mathcal{Q} : |\overline{f}(p)| > \delta$) and we reduce the set of centers $\mathcal{C} \cup \mathcal{Q}$ by using SIMS. In this way, at each step the computational cost for building the coefficient matrix in (9) and (14) depends on the number of points in $\mathcal{C} \cup \mathcal{Q}$ and it enables to achieve the required approximation accuracy without constructing the matrix K or K^TK related to the whole data set.

4 Discussion

In the previous sections, we defined a *top-down* approach to sparse approximation that fixes the centers of the RBFs and then computes the meaningful ones, with respect to ϵ, through an iterative procedure. The following discusses the numerical aspects of SIMS.

Initial set of input centers. Let us suppose that $y_i := 0$ and x_i is a point of the input data set, $i = 1, \dots, N$; then, the surface must interpolate the boundary constraints $f(x_i) = 0$ at $\mathcal{P}$. As done in [39] and in order to avoid the trivial solution $f \equiv 0$, we add positive (resp., negative)-valued normal constraints at x_i close to the boundary constraints and in the direction n_i (resp., $-n_i$), where n_i is the normal at x_i. If a polyhedral surface is given, surface normals are approximated by averaging the face normals; otherwise, we use those ones provided by the shape acquisition or achieved by a local fitting or clustering [1, 23]. Clearly, positive $\mathcal{P}^+$ and negative-valued $\mathcal{P}^-$ constraints can be set on a subset of $\mathcal{P}$ where shape variations occur and selected by using the Principal Components Analysis [21].

Therefore, as centers of the basis functions we can consider the set

1. $\mathcal{P} \cup \mathcal{P}^+ \cup \mathcal{P}^-$, thus solving the $3N \times 3N$ equation in (11);

2. $\mathcal{P}$ and solving the $N \times N$ least-squares problem (14).

The main differences between (9) and (14) are the size of the problem that has to be solved and the approximation error; sparse approximations obtained by using $3N$ centers guarantee a better accuracy with respect to the use of N centers, while requiring a greater computational cost for building and solving the associated linear system. Generally, the least-squares approximation is less sensible to noise or fine details in the input data; for instance, we can compare Figure 6(a-d) with (e).

Choosing the kernel function. If we select compactly supported RBFs, the least-squares approximation does not guarantee that the sparsity of K will be maintained in K^TK. Therefore, for large data sets the QR factorization of the sparse matrix K is evaluated at the first step and then used at each iteration to solve the linear system (14). Another choice is described in the paragraph "*Solving the linear system*" at page 9.

As sparse kernels we used $K(r) := (1-r)_+^4(4r+1)$ [40] and $K(r) := (1-r)_+^4(4+16r+12r^2+3r^3)$ [24, 32], where $r = \frac{\|x-y\|}{\sigma}$ and σ is its compact support. In both

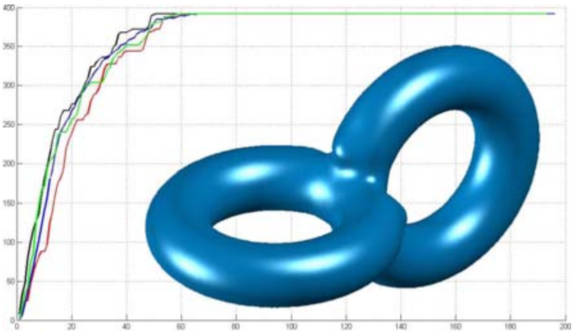


Figure 7. Surface reconstruction and convergence of the iterative sparse approximation with different initial guesses. The red line (resp., black, blue, and green) is related to $a^{(0)} = 0$ (resp., 1, random vector, and $K^{-1}y$).

Table 1. Computational cost of the iterative and least-squares sparse approximation.

Iterative Sparse Approximation

Task	Computational Cost
kd-Tree	$O(N \log N)$
Matrix Update	$O(N)$
Sol. lin. syst.	$O(N)$ iter. solv.

Least-squares Sparse Approximation

Task	Computational Cost
kd-Tree	$O(N \log N)$
Matrix Update	$O(N)$
Sol. lin. syst.	$O(N)$ iter. sol.
Sol. lin. syst.	$O(N^2)$ direct sol.

cases, the kernel belongs to $C^2(\mathbb{R}^3)$ and the corresponding sparse matrix is built by using a kd-Tree [5].

Initial guess choice. The initial point $a^{(0)}$ can be arbitrarily chosen or set to the *optimal guess* $K^{-1}y$ (resp., $K^{\dagger}y$ if we consider the least-squares approach), which corresponds to the solution of (9) (resp., (14)) if $\epsilon = 0$. This last choice guarantees a lower number of iterations but requires an additional computational cost for its evaluation. In the example shown in Figure 7, the iteration becomes stationary after 95 steps if $a^{(0)} = K^{-1}y$ (green line), requires 127 iterations if $a^{(0)} = 1$ (black line), and converges after 147 steps if the initial guess is randomly selected (blue line). The choice $a^{(0)} = 0$ can make unstable the initial iterations even though it does not affect the convergence of the method; this phenomenon is mainly due to the perturbation of the main diagonal of K in (11) with the factor $\epsilon/\sqrt{\eta}$, which tends to $+\infty$ when $\eta \rightarrow 0^+$. In the examples of the paper, we used $a^{(0)} = 1$ and $\eta = 10^{-20}$.

We conclude that the number of iterations required to have the convergence of Θ to $N - \overline{n}$ depends mainly on the eigenvalues of the sequence of matrices $K + \epsilon\Delta(a^{(n)})$, $n \in \mathbb{N}$, while the initial guess $a^{(0)}$ has a lower influence on the convergence speed. In fact, some choices of $a^{(0)}$ can reduce the number of iterations but slightly affect the number of null coefficients during the iterations.

Coefficient matrix updates and stop criteria. At step n, we update only the diagonal of K in $O(N)$-time (see Equation (11)) without affecting its sparsity percentage. The iteration of the algorithm stops when the number of null elements in $a^{(n)}$ becomes stationary, or when the residual error between two consecutive iterations is below a given threshold δ, *i.e.* $\|a^{(n+1)} - a^{(n)}\|_\infty \leq \delta$.

Solving the linear system. At each iteration n, $a^{(n)}$ has to be efficiently evaluated by solving the $D \times D$ linear system

$$[A + \epsilon\Delta(a^{(n)})]a^{(n+1)} = b$$

where (A, b) is (K, y) or $(K^T K, K^T y)$ depending on the type of sparse approximation we have taken into account, and by exploiting the specific structure or property (*e.g.*, sparsity, positive definiteness) of K. Here, D is the dimension of the problem, *i.e.*, N or $3N$.

If the coefficient matrix is sparse, iterative solvers such as the Gauss-Seidel (12) and bi-conjugate gradients stabilized method [19] have a linear computational cost. If A is positive definite, another way to proceed consists of using its Cholesky factorization, which can be efficiently updated at each iteration when A is replaced by $A + xx^T$, $x \in \mathbb{R}^D$. To this end, we note that

$$A^{(n)} := A + \epsilon\Delta(a^{(n)}) = A + \sum_{i=1}^{D} x_i^{(n)} x_i^{(n)T},$$

$$(x_i^{(n)})_j := \left\{ \begin{array}{ll} 0 & j \neq i, \\ \sqrt{\epsilon}\delta_{ii}^{(n)} & j = i, \end{array} \right.$$

$\delta_{ii}^{(n)} := (a_i^{(n)} + \eta)^{-1/4}$, and therefore the Cholesky factor of $A^{(n)}$ is achieved after D updates to the Cholesky factor of A.

If A is semi-positive definite (*e.g.*, $A := K^T K$) we consider the Cholesky factorization $A^{(1)} = C^T C$ of the positive-definite matrix $A^{(1)}$ and we use it to find the Cholesky factor of $A^{(n)}$, $n \geq 1$. In this case, we use D iterations to add to $A^{(1)}$ the diagonal matrix $\Delta(a^{(n-1)})$, and

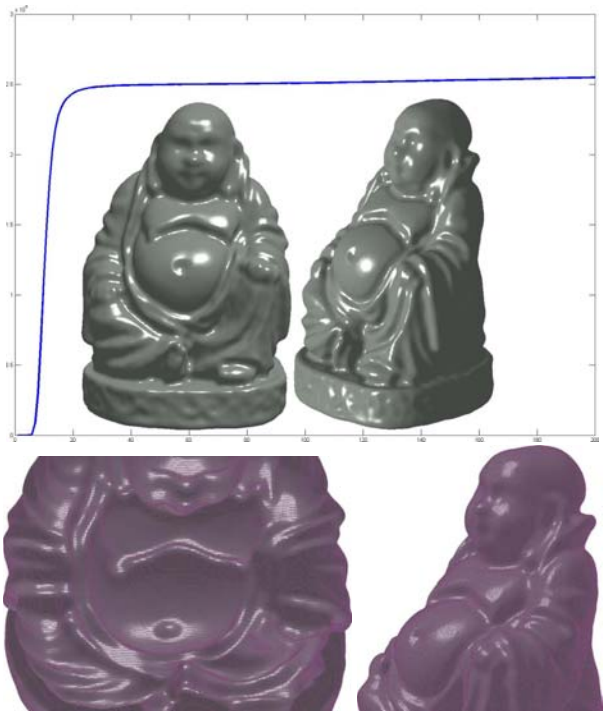


Figure 8. Iterative sparse approximation and reconstruction; with compactly supported basis functions. $25K$ centers on $78K$ have been discarded in the reconstruction.

D iterations to remove $\Delta(a^{(1)})$ while updating C. The previous steps cannot be reversed because the Cholesky factorization, as well its updates, always requires to deal with positive definite matrices.

Table 1 summarizes the computational cost of the main steps of the iterative and least-squares sparse approximation. Additional examples and timings are shown in Figures 8-10 and Table 2.

5 Conclusions and Future work

While there has been much work on implicit modeling and related applications, the study of implicit sparse approximations has been focused mainly on Support Vector Machine. Exploiting a formulation equivalent to SVM and based on Tikhonov regularization, the paper discussed an iterative method that defines a compact surface approximation while guaranteeing a good approximation accuracy. We introduced a novel multi-level sparsification scheme where a different resolution in the approximation of the input data is achieved by selecting the parameter ϵ or a different number of centers, which have been sorted in increasing order of importance by the iterative procedure.

The advantages of our approach are its efficiency and simplicity; it differs from previous work on the use of an approximated formulation of the sparsification problem, which avoids to use heuristic solvers of convex quadratic optimization problems. This aim is simply achieved by taking advantage of a smooth approximation of the l_1-norm and of robust iterative solvers of systems of non-linear equations.

Even thought the number of used centers can be set in a simple way trough the multi-level scheme, the main open issue is to control directly, and not through the parameter ϵ, the final error approximation.

Acknowledgments.

Special thanks are given to Alexander Belyaev and the AG4-Computer Graphics Group at MPII-Saarbruecken for motivating and supporting the author to consider the problem discussed here. We thank the anonymous reviewers for their constructive suggestions and the Shape Modeling Group at IMATI-CNR. This work has been supported by the EC-IST FP6 Network of Excellence “AIM@SHAPE”; models are courtesy of the AIM@SHAPE repository (*http://shapes.aimatshape.net*).

References

[1] M. Alexa, J. Behr, D. Cohen-Or, S. Fleishman, D. Levin, and C. T. Silva. Computing and rendering point set surfaces. *IEEE Trans. Vis. Comput. Graph*, 9(1):3–15, 2003.

[2] N. Amenta, M. Bern, and M. Kamvysselis. A new voronoi-based surface reconstruction algorithm. In *SIGGRAPH-98*, pages 415–421, 1998.

[3] A. Angelidis and M.-P. Cani. Adaptive implicit modeling using subdivision curves and surfaces as skeletons. In *SMA '02: Proc. of the ACM Symposium on Solid modeling and Applications*, pages 45–52. ACM Press, 2002.

[4] N. Aronszajn. Theory of reproducing kernels. *Trans. Amer. Math. Soc.*, 68:337–404, 1950.

[5] S. Arya, D. Mount, N. Netanyahu, R. Silverman, and A. Wu. An optimal algorithm for approximate nearest neighbor searching fixed dimensions. *JACM: Journal of the ACM*, 45, 1998.

[6] M. Bertero. *Regularization methods for linear inverse problems*. Inverse Problems, Springer-Verlag, Berlin, 1986.

[7] J. Bloomenthal and B. Wyvill, editors. *Introduction to Implicit Surfaces*. Morgan Kaufmann Publishers Inc., 1997.

[8] M. Botsch and L. Kobbelt. Real-time shape editing using radial basis functions. *Eurographics*, 3:611–621, 2005.

[9] J. C. Carr, R. K. Beatson, J. B. Cherrie, T. J. Mitchell, W. R. Fright, B. C. McCallum, and T. R. Evans. Reconstruction and representation of 3d objects with radial basis functions. In *SIGGRAPH '01*, pages 67–76. ACM Press, 2001.

Table 2. The table shows the input parameters, timings, and number of iterations used by SIMS for the examples of the paper; tests are performed on a Pentium IV 2.80 GHz.

Test	N	k-neighb.	Matrix Comput. [sec]	ϵ	♯Iter.	Iter. Time [sec.]	%Sel. centers	L^∞-error
Figure 2(c)	$240K$	10	7.49	1.2	22	15.68	62%	3.1×10^{-4}
Figure 3(b)	$240K$	20	12.69	1.2	48	41.03	60%	5.2×10^{-6}
Figure 8	$75K$	20	3.79	0.5	65	16.70	68%	1.52×10^{-4}
Figure 9	$78K$	20	3.49	0.5	69	17.51	60%	2.31×10^{-4}
Figure 10	$78K$	10	1.91	0.8	36	7.38	64%	3.6×10^{-5}

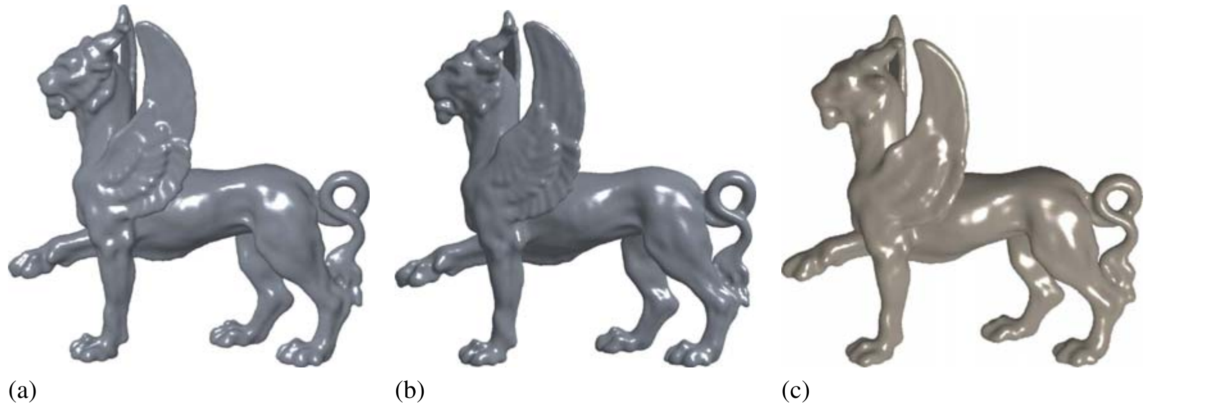


Figure 9. Reconstruction achieved with different percentages of selected centers: (a) 87.39%**, (b)** 66.45%**, (c)** 47.21% **on 150K input centers.**

[10] S. Chen and J. Wigger. Fast orthogonal least squares algorithm for efficient subset model selection. *IEEE Transactions on Signal Processing*, 43(7):1713–1715, July 1995.

[11] S. S. Chen, D. L. Donoho, and M. A. Saunders. Atomic decomposition by basis pursuit. *SIAM Journal on Scientific Computing*, 20(1):33–61, 1998.

[12] T. F. Coleman and Y. Li. An interior trust region approach for nonlinear minimization subject to bounds. *SIAM Journal on Optimization*, 6:418–445, 1996.

[13] C. Cortes and V. Vapnik. Support-vector networks. *Machine Learning*, 20(3):273–297, 1995.

[14] T. K. Dey and S. Goswami. Tight cocone: a water-tight surface reconstructor. In *Symposium on Solid Modeling and Applications*, pages 127–134, 2003.

[15] M. Fenn and S. Gabriele. Robust local approximation of scattered data. *Geometric Properties from Incomplete Data*, 45, 1998.

[16] S. Fleishman, D. Cohen-Or, M. Alexa, and C. T. Silva. Progressive point set surfaces. *ACM Trans. Graph*, 22(4):997–1011, 2003.

[17] S. Fleishman, D. Cohen-Or, and C. T. Silva. Robust moving least-squares fitting with sharp features. *ACM Trans. Graph*, 24(3), 2005.

[18] F. Girosi. An equivalence between sparse approximation and support vector machines. *Neural Computation*, 10(6):1455–1480, 1998.

[19] G. Golub and G. VanLoan. *Matrix Computations*. John Hopkins University Press, 2nd. edition, 1989.

[20] H. Hoppe. Progressive meshes. In *SIGGRAPH '96 Proc.*, pages 99–108, Aug. 1996.

[21] I. T. Jolliffe. Principal component analysis. In *Principal Component Analysis*. Springer Verlag, 1986.

[22] M. M. Kazhdan. Reconstruction of solid models from oriented point sets. In *Symposium on Geometry Processing*, pages 73–82, 2005.

[23] D. Levin. The approximation power of moving least-squares. *Math. Comput*, 67(224):1517–1531, 1998.

[24] B. S. Morse, T. S. Yoo, D. T. Chen, P. Rheingans, and K. R. Subramanian. Interpolating implicit surfaces from scattered surface data using compactly supported radial basis functions. In *Proc. of SMI*, pages 89–98. IEEE Computer Society, 2001.

[25] Y. Ohtake, A. Belyaev, M. Alexa, G. Turk, and H.-P. Seidel. Multi-level partition of unity implicits. *ACM Trans. Graph.*, 22(3):463–470, 2003.

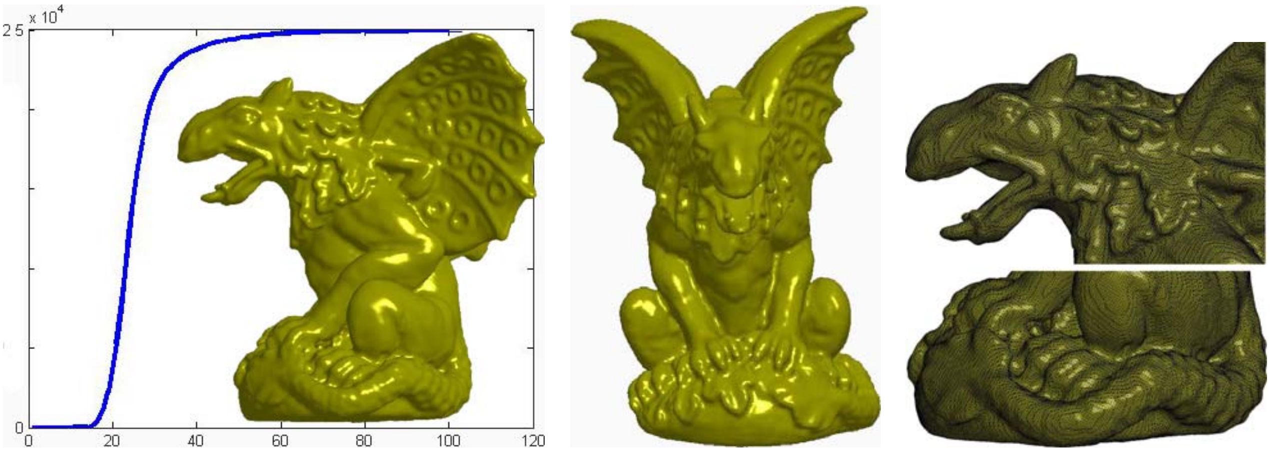


Figure 10. Sparsification and reconstruction using $48K$ centers on $75K$.

[26] Y. Ohtake, A. Belyaev, and H.-P. Seidel. 3d scattered data interpolation and approximation with multilevel compactly supported RBFs. *Graph. Models*, 67(3):150–165, 2005.

[27] Y. Ohtake, A. G. Belyaev, and M. Alexa. Sparse low-degree implicits with applications to high quality rendering, feature extraction, and smoothing. In *Symposium on Geometry Processing*, pages 149–158, 2005.

[28] Y. Ohtake, A. G. Belyaev, and H.-P. Seidel. 3D scattered data approximation with adaptive compactly supported radial basis functions. In *Proc. of SMI*, pages 31–39, 2004.

[29] M. Pauly, M. H. Gross, and L. Kobbelt. Efficient simplification of point-sampled surfaces. In *IEEE Visualization*, 2002.

[30] M. Pauly, R. Keiser, L. P. Kobbelt, and M. Gross. Shape modeling with point-sampled geometry. In J. Hodgins and J. C. Hart, editors, *Proceedings of ACM SIGGRAPH 2003*, volume 22(3), pages 641–650. ACM Press, 2003.

[31] B. Schoelkopf and A. J. Smola. *Learning with Kernels*. The MIT Press, 2002.

[32] B. Schoelkopf, F. Steinke, and V. Blanz. Object correspondence as a machine learning problem. In *ICML '05: Proceedings of the 22nd international conference on Machine learning*, pages 776–783. ACM Press, 2005.

[33] B. Schölkopf, J. Giesen, and S. Spalinger. Kernel methods for implicit surface modeling. In L. K. Saul, Y. Weiss, and L. Bottou, editors, *Advances in Neural Information Processing Systems 17*, pages 1193–1200. MIT Press, Cambridge, MA, 2005.

[34] C. Shen, J. F. O'Brien, and J. R. Shewchuk. Interpolating and approximating implicit surfaces from polygon soup. *ACM Trans. Graph*, 23(3):896–904, 2004.

[35] F. Steinke, B. Schoelkopf, and V. Blanz. Support vector machines for 3d shape processing. In *Proc. EUROGRAPHICS*, volume 24 (3), pages 285–294, 2005.

[36] A. Tikhonov and V. Arsenin. *Solutions of Ill-posed Problems*. W.H. Winston Washington, D.C., 1977.

[37] I. Tobor, P. Reuter, and C. Schlick. Multi-scale reconstruction of implicit surfaces with attributes from large unorganized point sets. In *Proc. of SMI*, pages 19–30, 2004.

[38] G. Turk and J. F. O'Brien. Shape transformation using variational implicit functions. In *Proc. of SIGGRAPH '99*, pages 335–342. ACM Press/Addison-Wesley Publishing Co., 1999.

[39] G. Turk and J. F. O'Brien. Modelling with implicit surfaces that interpolate. *ACM Trans. Graph.*, 21(4):855–873, 2002.

[40] H. Wendland. Real piecewise polynomial, positive definite and compactly supported radial functions of minimal degree. *Adv. Comput. Math.*, 4(4):389–396, 1995.

[41] H. Xie, K. T. McDonnell, and H. Qin. Surface reconstruction of noisy and defective data sets. In *IEEE Visualization*, pages 259–266, 2004.