Local Approximation of Scalar Functions on 3D Shapes and Volumetric Data

Giuseppe Patanè, Michela Spagnuolo
CNR-IMATI, Genova−Italy

Abstract

In this paper, we tackle the problem of computing a map that locally interpolates or approximates the values of a scalar function, which have been sampled on a surface or a volumetric domain. We propose a local approximation with radial basis functions, which conjugates different features such as locality, independence of any tessellation of the sample points, and approximation accuracy. The proposed approach handles maps defined on both 3D shapes and volumetric data and has extrapolation capabilities higher than linear precision methods and moving least-squares techniques with polynomial functions. It is also robust with respect to data discretization and computationally efficient through the solution of a small and well-conditioned linear system. With respect to previous work, it allows an easy control on the preservation of local details and smoothness through both interpolating and least-squares constraints. The main application we consider, is the approximation of maps defined on grids, 3D shapes, and volumetric data.

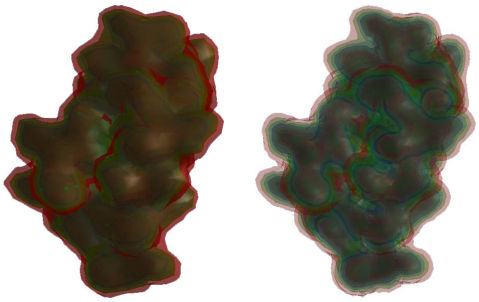
1. Introduction

Scalar functions on 3D shapes are capable of capturing properties relevant to the definition of shape descriptors and approximation schemes have been recently used to extend these functions from the input shape to another surface in order to identify corresponding feature elements [1] or compute harmonic volumetric mappings [2, 3, 4, 5]. Addressing the computation of volumetric [6] or template-based [7] shape descriptors by extending the surface-based Laplacian eigenvectors to the surrounding volume [8] avoids the evaluation of volumetric descriptors directly on a tessellation of the input shape. In this way, the computational cost, which is generally high in case of volumetric meshes, is effectively reduced. Furthermore, descriptors primarily defined on surfaces (e.g., GPS embedding [9], biharmonic distances [10, 8], heat kernel signatures [11]) are approximated to code volumetric information.

Our work tackles the problem of computing the map $F : \mathbb{R}^3 \to \mathbb{R}$ underlying a discrete scalar function $f : \mathcal{P} \to \mathbb{R}$, defined on a set $\mathcal{P} := \{\mathbf{x}_i\}_{i=1}^n$ of points in $\mathbb{R}^3$, which have been sampled on a surface or a volumetric domain. The map F is defined as a function that *locally* interpolates or approximates the f-values.

The idea behind the proposed approach is to compute $F(\mathbf{x})$, $\mathbf{x} \in \mathbb{R}^3$, by imposing the f-values at the points of a neighbor $\mathcal{N}_\mathbf{x} := \{\mathbf{x}_{j_s}\}_{s=1}^k$ of $\mathbf{x}$ as interpolating or least-squares (LS) constraints. This choice is motivated by the observation that the behavior of any approximation of f at $\mathbf{x}$ is mainly controlled by the f-values in $\mathcal{N}_\mathbf{x}$. Since the value $F(\mathbf{x})$ is computed through the solution of a $k \times k$ linear system, where k is generally small (i.e., $20 \le k \le 30$) and much lower than n, the cost for its evaluation varies from $O(k^3)$ to $O(k)$, according to the sparsity of the coefficient matrix and the use of direct or iterative solvers [12]. Dealing with a small linear system also guarantees a lower conditioning number, which is proportional to the size of the coefficient matrix and a lower computational cost for the use of preconditioners.

Global approximation schemes [13, 14, 15, 16, 17] apply interpolating or LS constraints *globally*; then, the resulting approximation is evaluated at any sample point. Since a $n \times n$ linear system is solved once, the computational cost of the approximation with globally- and locally-supported RBFs is $O(n^3)$ and $O(n \log n)$, respectively. In contrast, we solve a linear system once for each sample point and the evaluation of the resulting approximation at s points takes $O(sk^3)$ time. Indeed, its computational cost is generally lower than the approximation with globally- and locally-supported RBFs. It also reduces the memory storage from $O(n^3)$ and $O(kn)$ to $O(k^3)$. Finally, our method has the same order of complexity of local approximation schemes, such as the moving least-squares (MLS) approximation [13, 14, 15]



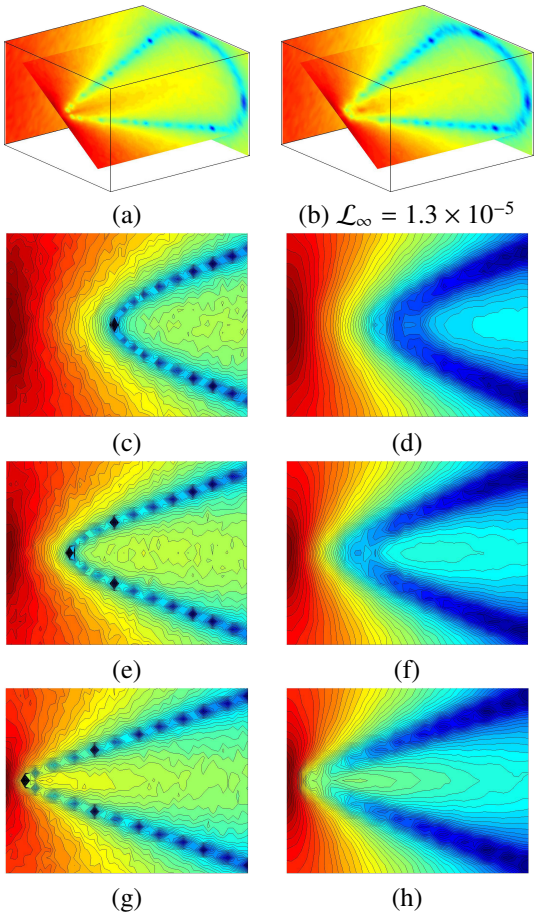
(a) $e := 64$ (b) $e_{new} := 256$ $\mathcal{L}_\infty = 2.3 \times 10^{-7}$

Figure 1: (a) Different iso-surfaces of the discrete map $f : \mathcal{P} \to \mathbb{R}$, achieved by sampling the electrostatic charge of a molecule on a low resolution grid $\mathcal{P}$ ($e \times e \times e$). (b) Iso-surfaces of the corresponding local approximation $F : \mathbb{R}^3 \to \mathbb{R}$ of $\{f(\mathbf{x})\}_{\mathbf{x}\in\mathcal{P}}$ on a $e_{new} \times e_{new} \times e_{new}$ grid $\mathcal{S}$.

and the multi-level Partition of the Unity (PU) [18].

Since MLS approximations [13, 14, 15] and the multi-level PU [18] involve a polynomial basis, they cannot interpolate the f-values in a simple way. In fact, the degree of the fitting polynomial determines the number of interpolating conditions and not *vice-versa*. For instance, in 3D a polynomial of degree two or three requires to impose ten or nineteen interpolating constraints; however, we might have a different number of points in different neighbors. Furthermore, in case of uneven sampling fixing the number of points in each neighbor instead of its radius, or increasing the polynomial degree, provides unstable results due to the ill-conditioning of the corresponding Gram matrices [12]. Using a set of radial instead of polynomial basis functions allows us to combine interpolating constraints for feature preservation and LS conditions for noise removal. In fact, the number of local interpolating constraints is equal to the number of RBFs and no more related to the degree of the polynomial used for the local approximation. In this way, we improve the flexibility in the design of scalar functions with sparse constraints [19], which uses the PU and reproduces only linear maps.

While approximation schemes have been specialized for scalar functions defined on surfaces [20, 6], or point sets [21], or volumetric grids [22, 23, 24, 25, 26, 27], the local approximation handles scalar functions defined on grids, 3D shapes, and volumes. Furthermore, it is not restricted to the preservation of the critical points of the input map [28] and its approximation accuracy is higher than linear precision methods [6] and MLS techniques



(a) (b) $\mathcal{L}_\infty = 1.3 \times 10^{-5}$ (c) (d) (e) (f) (g) (h)

Figure 2: (a) Input noisy map $f : \mathcal{P} \to \mathbb{R}$ that represents the flow values on a voxel grid $\mathcal{P}$ and (b) corresponding local LS approximation. Level-sets of (c,e,g) f and (d,f,h) its approximation.

with polynomial functions. While previous work generally requires a parameterization domain [29, 4, 5, 30], the proposed approximation uses only the function values and does not require an underlying tessellation of $\mathcal{P}$.

Our tests on surface reconstruction have shown that the proposed approach is useful for approximating point sets that are noisy or have been sampled at low resolution. In fact, the local LS approximation with RBFs reduces the number of local artifacts, which are typically introduced where a coarse discretization of the normals is used to recover the interior and exterior part of the reconstructed surface. For densely sampled shapes, polynomial MLS surfaces already provide satisfactory results in terms of feature preservation, accuracy, and computational cost. Therefore, the local LS approximation with RBFs improves the approximation accuracy at

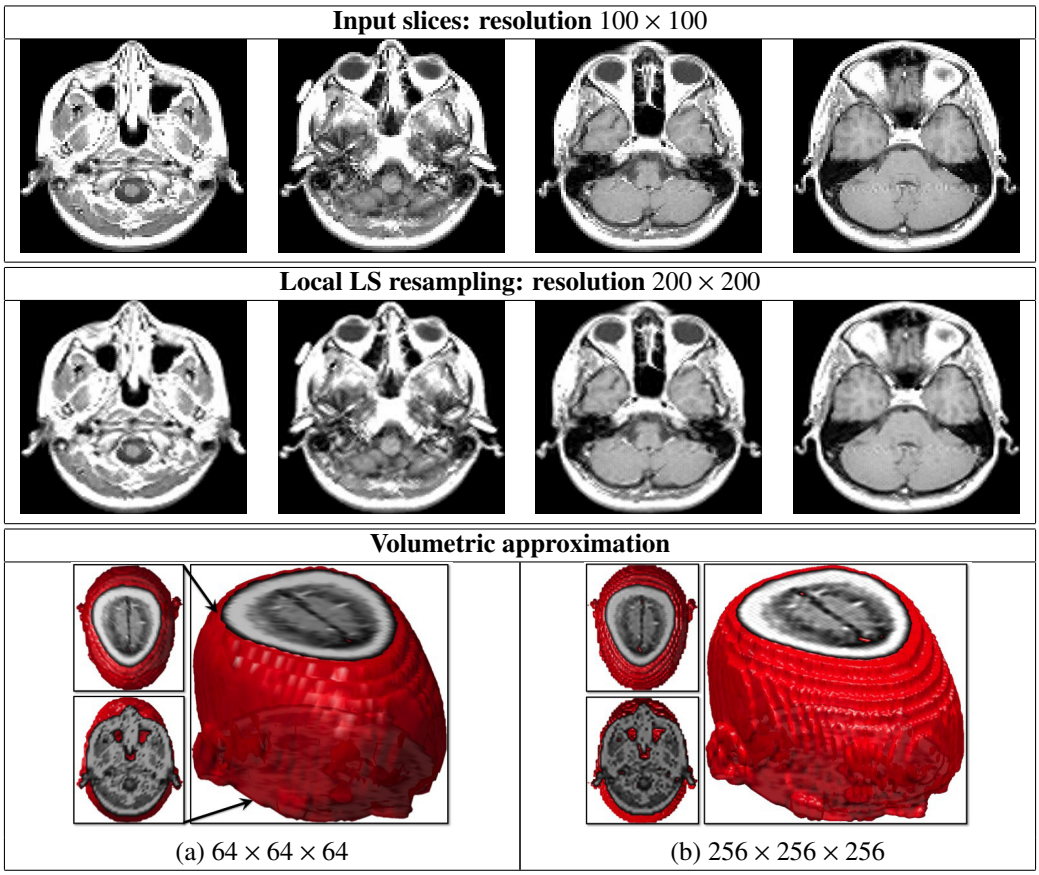


Figure 3: (Second row) Local LS approximation of (first row) a low-resolution and noisy medical data set. Iso-surfaces related to the (a) input and (b) resampled data set.

the price of a higher computational cost. Finally, tests on image upsampling have shown that our method is better suited to extrapolate local information. In fact, the redundant information available in high-resolution images is effectively removed with linear downsampling methods, without applying accurate schemes such as the local approximation with RBFs.

As an overview of our approach, the energy of the electrostatic charge (Fig. 1) of a molecule and of a flow field (Fig. 2) have been measured at the nodes of a coarse grid in $\mathbb{R}^3$ and the corresponding local approximations have been resampled on a finer grid. The normalized $\mathcal{L}_\infty$ approximation error $\|F - f\|_\infty / \|f\|_\infty$ confirms the accuracy of the proposed approach. In Fig. 3, the local approximation with LS constraints of 2D images have been resampled at a higher resolution. Complex features, which are located where different tissues and bones touch each others, are accurately approximated.

The paper is organized as follows. We introduce the local (Sect. 2) and spectral volumetric approximation (Sect. 3); then, we outline future work (Sect. 4).

2. Local approximation with RBFs

Firstly, we define the function $F : \mathbb{R}^3 \to \mathbb{R}$ underlying the discrete map $f : \mathcal{P} \to \mathbb{R}$ through a local approximation scheme with interpolating and LS constraints (Sect. 2.1). Then, we discuss its main properties (Sect. 2.2) and applications (Sect. 2.3).

2.1. Local approximation with interpolating and least-squares constraints

First of all, each point $\mathbf{x} \in \mathbb{R}^3$ is associated to the neighbor $\mathcal{N}_\mathbf{x} := \{\mathbf{x}_{j_s}\}_{s=1}^k$ of $\mathbf{x}$, which includes those points of $\mathcal{P}$ that fall inside the sphere of center $\mathbf{x}$ and radius $\sigma(\mathbf{x})$; i.e., $\|\mathbf{x}_{j_s} - \mathbf{x}\|_2 \leq \sigma(\mathbf{x})$, $s = 1, \ldots, k$ (Fig. 4(a)). Here, the value $\sigma(\mathbf{x})$ is chosen according to the local sampling density of $\mathcal{P}$ [31]. For simplicity, we omit the dependence of the number k of

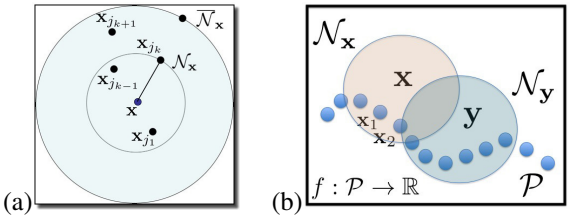


Figure 4: (a,b) Neighbors $\mathcal{N}_{\mathbf{x}}$ and $\overline{\mathcal{N}}_{\mathbf{x}}$ of $\mathbf{x}$, $\mathcal{N}_{\mathbf{x}} \subseteq \overline{\mathcal{N}}_{\mathbf{x}}$, used for the computation of the value $F(\mathbf{x})$.

points in $\mathcal{N}_{\mathbf{x}}$ from $\mathbf{x}$ and $\sigma(\mathbf{x})$. In $\mathcal{N}_{\mathbf{x}}$, we approximate the input scalar function $f : \mathcal{P} \to \mathbb{R}$ with the implicit map $F : \mathbb{R}^3 \to \mathbb{R}$, which is the linear combination of the RBFs $\mathcal{B} := \{\phi_{j_s}(\mathbf{x}) := \phi(\|\mathbf{x} - \mathbf{x}_{j_s}\|_2)\}_{s=1}^{k}$; i.e.,

$$\begin{cases} F(\mathbf{x}) := \sum_{s=1}^{k} \beta_s(\mathbf{x})\phi_{j_s}(\mathbf{x}) = \beta^T(\mathbf{x})\tilde{\phi}(\mathbf{x}), \\ \beta(\mathbf{x}) := (\beta_s(\mathbf{x}))_{s=1}^{k}, \quad \tilde{\phi}(\mathbf{x}) := (\phi_{j_s}(\mathbf{x}))_{s=1}^{k}. \end{cases} \tag{1}$$

Each function ϕ_{j_s} is generated by a map $\phi : \mathbb{R}^+ \to \mathbb{R}$ and centered at $\mathbf{x}_{j_s}$ [13, 14]. We also assume that the corresponding kernel $K(\mathbf{x}, \mathbf{y}) := \phi(\|\mathbf{x} - \mathbf{y}\|_2^2)$, $\mathbf{x}, \mathbf{y} \in \mathbb{R}^3$, is positive definite; i.e., the Gram matrix associated to $K(\cdot, \cdot)$ is positive definite.

Local approximation with interpolating constraints. For the computation of the value $F(\mathbf{x})$, as interpolating conditions we select the f-values at those points of $\mathcal{P}$ that belong to the neighbor $\mathcal{N}_{\mathbf{x}}$ of the evaluation point $\mathbf{x}$ instead of the whole set of function values. This choice is motivated by the observation that the behavior of any approximation of f at $\mathbf{x}$ is mainly controlled by the f-values in $\mathcal{N}_{\mathbf{x}}$. Indeed, in $\mathcal{N}_{\mathbf{x}}$ we compute the function $F : \mathbb{R}^3 \to \mathbb{R}$ that interpolates the values of f at the points of $\mathcal{N}_{\mathbf{x}}$; i.e., $F(\mathbf{x}_{j_s}) = f(\mathbf{x}_{j_s})$, $s = 1, \ldots, k$.

Imposing these interpolating conditions, the coefficient vector $\beta(\mathbf{x})$ in Eq. (1) solves the linear system

$$\Phi\beta(\mathbf{x}) = \mathbf{f}_k, \qquad \Phi := (\phi_{rs})_{r,s=1}^{k},$$

where Φ is the $k \times k$ Gram matrix associated to the generating map ϕ with respect to the points of $\mathcal{N}_{\mathbf{x}}$, whose entries are $\phi_{rs} := \phi(\|\mathbf{x}_{j_r} - \mathbf{x}_{j_s}\|_2)$, and $\mathbf{f}_k := (f(\mathbf{x}_{j_s}))_{s=1}^{k}$ is the $k \times 1$ array of the corresponding function values.

The novelty of the proposed approach is to apply this scheme *locally*; for the evaluation of $F(\mathbf{x})$, we consider as interpolating constraints only the f-values at those points of $\mathcal{P}$ that belong to $\mathcal{N}_{\mathbf{x}}$. Indeed, the evaluation point $\mathbf{x}$ drives the selection of a set of interpolating conditions and the construction of the local approximation F in $\mathcal{N}_{\mathbf{x}}$ (Fig. 4(b)). On the contrary, previous work applies these constraints *globally*; i.e., the function $F : \mathbb{R}^3 \to \mathbb{R}$ that satisfies *all* the interpolating conditions $F(\mathbf{x}_i) = f(\mathbf{x}_i)$, $i = 1, \ldots, n$, is computed and then F is evaluated at any point $\mathbf{x}$.

Algorithm 1: Main steps of the evaluation of the local approximation $F(\mathbf{x})$ at a sample point $\mathbf{x}$ (Fig. 4(b)).

Require: A point set $\mathcal{P} := \{\mathbf{x}_i\}_{i=1}^{n}$, a scalar function $f : \mathcal{P} \to \mathbb{R}$, and a set $\mathcal{S}$ of sample points.
Ensure: The value $F(\mathbf{x})$, $\mathbf{x} \in \mathcal{S}$, where $F : \mathbb{R}^3 \to \mathbb{R}$ is the local approximation of f with LS constraints.
1: Compute the nearest neighbor graph of $\mathcal{P}$ [32].
2: **for** each evaluation point $\mathbf{x} \in \mathcal{S}$ **do**
3: compute the neighbors $\mathcal{N}_{\mathbf{x}}$, $\overline{\mathcal{N}}_{\mathbf{x}}$: $\mathcal{N}_{\mathbf{x}} \subseteq \overline{\mathcal{N}}_{\mathbf{x}}$;
4: compute the Gram matrix Φ in Eq. (2);
5: compute the vector $\mathbf{f}_h$ in Eq. (2);
6: compute $\beta(\mathbf{x})$ through Eq. (2);
7: evaluate $F(\mathbf{x})$ in Eq. (1).
8: **end for**

Local approximation with least-squares constraints. Interpolating the f-values allows us to precisely reproduce the local behavior of f but is error-prone in case of noisy data. To overcome this drawback, we replace interpolating with LS constraints. Recalling that in $\mathcal{N}_{\mathbf{x}}$ the function $F : \mathbb{R}^3 \to \mathbb{R}$ underlying $f : \mathcal{P} \to \mathbb{R}$ is a sum of RBFs centered at the points of $\mathcal{N}_{\mathbf{x}}$, the local LS approximation takes into account the f-values at the points of a neighbor $\overline{\mathcal{N}}_{\mathbf{x}}$ larger than $\mathcal{N}_{\mathbf{x}}$. In our tests, $\overline{\mathcal{N}}_{\mathbf{x}}$ is the neighbor of $\mathbf{x}$ whose radius is twice the radius of $\mathcal{N}_{\mathbf{x}}$. Then, F is computed by minimizing the error $\sum_{s=1}^{h} |F(\mathbf{x}_{j_s}) - f(\mathbf{x}_{j_s})|^2$ through the normal equation

$$(\Phi^T\Phi)\beta(\mathbf{x}) = \Phi^T\mathbf{f}_h, \quad \Phi := (\phi_{sr})_{s=1,\ldots,h}^{r=1,\ldots,k}, \tag{2}$$

where $\phi_{sr} := \phi(\|\mathbf{x}_{j_r} - \mathbf{x}_{j_s}\|_2)^2$, $\overline{\mathcal{N}}_{\mathbf{x}} := \{\mathbf{x}_{j_s}\}_{s=1}^{h}$, $k \le h \le n$, and $\mathbf{f}_h := (f(\mathbf{x}_{j_s}))_{s=1}^{h}$ is the $h \times 1$ right-hand side vector. Since Φ is a full-rank matrix, $(\Phi^T\Phi)$ is invertible and Eq. (2) has a unique solution. If $\mathcal{N}_{\mathbf{x}} = \overline{\mathcal{N}}_{\mathbf{x}}$, then the local approximations with interpolating and LS constraints are the same. The main steps for the evaluation of the local approximation at a sample point are summarized in Algorithm 1.

Examples. In Fig. 5, three smooth signals $\tilde{f} : \mathcal{I} \to \mathbb{R}$ have been sampled on a set $\mathcal{P}$ of 100 points of the interval $\mathcal{I} := [0, 4\pi]$, thus achieving the discrete signal $f : \mathcal{P} \to \mathbb{R}$, $\mathcal{P} := \{x_i\}_{i=1}^{n}$, $x_i < x_{i+1}$. Using f as input data, we have computed the local approximation $F : \mathbb{R} \to \mathbb{R}$ with RBFs, which is evaluated on $\mathcal{P}$ and a set $\mathcal{S}$ of samples finer than $\mathcal{P}$; i.e., $\mathcal{P} \subseteq \mathcal{S}$. Indicating with $\epsilon_{Q,\infty} := \max_{x_i \in Q}\{|F(x_i) - f(x_i)|\}$ the maximum

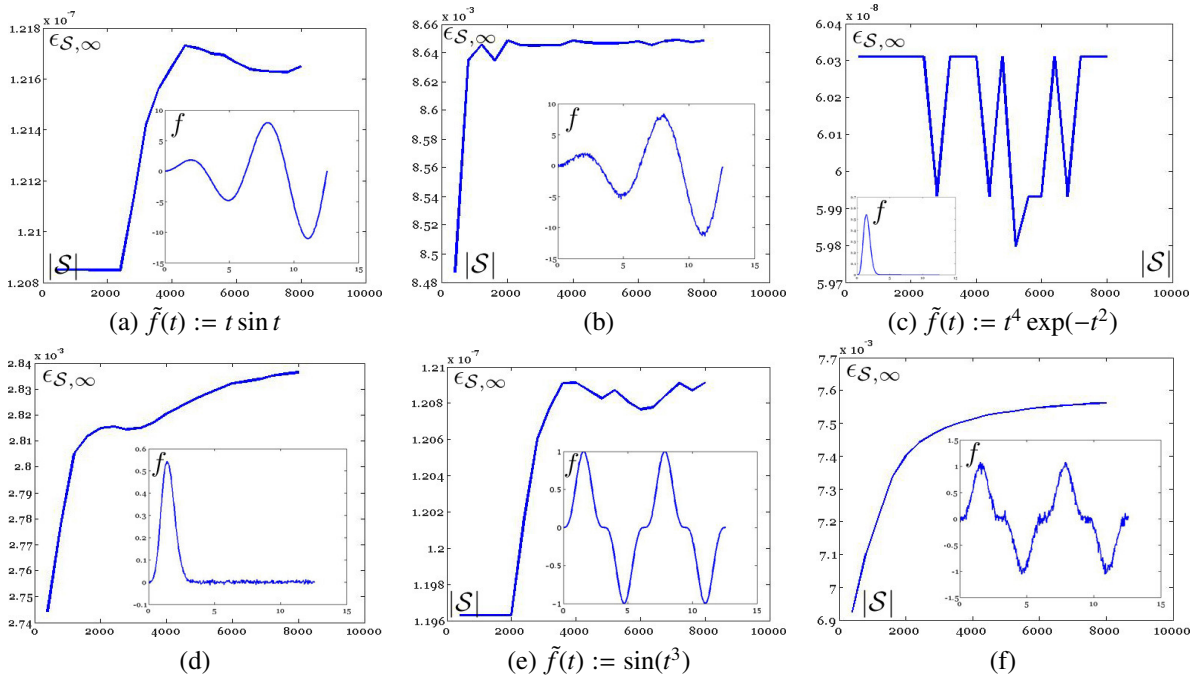


Figure 5: The smooth signal $\tilde{f} : \mathcal{I} \to \mathbb{R}$ has been sampled on a set $\mathcal{P}$ of 100 points of the interval $\mathcal{I} := [0, 4\pi]$, thus achieving the discrete map $f : \mathcal{P} \to \mathbb{R}$. The graphs in (a,c,e) show the extrapolation error $\epsilon_{\mathcal{S},\infty}$ (y-axis) between the local approximation $F : \mathbb{R} \to \mathbb{R}$ underlying f and the $\tilde{f}$-values on a set $\mathcal{S}$ of points, which includes $\mathcal{P}$ and has an increasing number (x-axis) of samples. The counterparts of these tests on noisy signals are shown in (b,d,f).

pointwise variation between f and F on a set $\mathcal{Q}$, we evaluate both the *approximation* and *extrapolation error* by selecting $\mathcal{Q} := \mathcal{P}$ and $\mathcal{Q} := \mathcal{S}$, respectively.

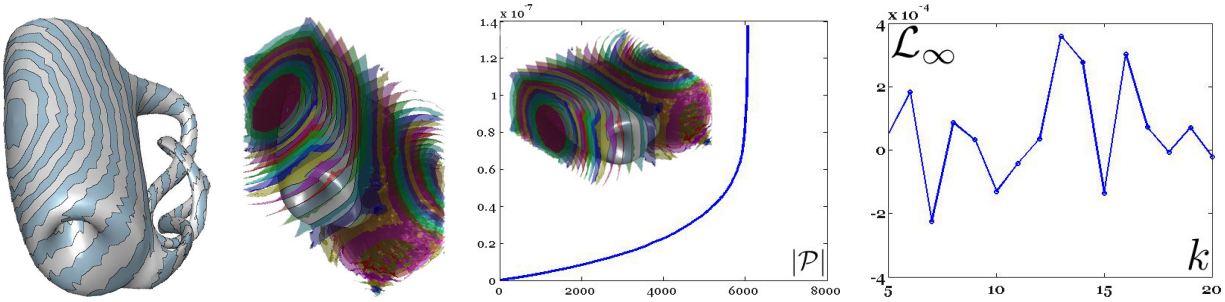
Due to the redundancy of the sampled values, the extrapolation error $\epsilon_{\mathcal{S},\infty}$ for the local approximation with interpolating constraints is lower than 10^{-7} (Fig. 5(a,c,e)). To appreciate the difference between interpolating and LS constraints, we have perturbed the input data (Fig. 5(a,c,e)) with a Gaussian noise (Fig. 5(b,d,f)); then, we have applied the local LS approximation. In this case, the extrapolation error remains lower than 10^{-2} and the local approximation with LS constraints attenuates the noise component, without deteriorating the approximation accuracy and the extrapolation capability.

Analogously to signal approximation, for 2D images the values of $f : \mathcal{P} \subseteq \mathbb{R}^2 \to \mathbb{R}$ are the pixels' intensity, the set $\mathcal{P}$ is the pixel grid, and the new sampling grid $\mathcal{S}$ is finer or coarser than $\mathcal{P}$ in case of up- or down-sampling, respectively. Due to the regularity of the image grid, $\mathcal{N}_{\mathbf{x}}$, $\overline{\mathcal{N}_{\mathbf{x}}}$ are the 4- and 8-connected neighbors of $\mathbf{x}$. To test the extrapolation capability, the local approximation underlying the input data in Fig. 6(a) is resampled on a higher resolution grid with interpolating (Fig. 6(b)) and LS (Fig. 6(c)) constraints. Increasing the neighbor size we get a larger number of basis functions in Eq. (1), which might introduce local and small perturbations to the evaluation of the F-values. As a result, the approximation error slightly increases but remains lower than 10^{-4} (Fig. 6(d)).

We now consider the values of a noisy map $f : \mathcal{P} \to \mathbb{R}$ (Fig. 7(a)), which has been computed by adding a Gaussian noise to the values of the volumetric map $\tilde{f} : \mathbb{R}^3 \to \mathbb{R}$ at the vertices $\mathcal{P}$ of a triangle mesh $\mathcal{M}$. The behavior of f and its underlying approximation F with LS constraints is visualized through the iso-contours $\{f^{-1}(\alpha)\}_{\alpha \in \mathbb{R}}$ and the iso-surfaces $\{F^{-1}(\alpha)\}_{\alpha \in \mathbb{R}}$, respectively. The smoothness of the iso-surfaces (Fig. 7(b,c)) confirms the regularity of F around $\mathcal{P}$. Finally, the $\mathcal{L}_\infty$ error (Fig. 7(c)) between $\tilde{f}$ and F on $\mathcal{P}$ is lower than 10^{-6}. Changes of the support size slightly affect the approximation accuracy of the local approximation. Increasing the neighbor radius forces the LS approximation (Fig. 7(d)) to use a larger number of constraints on the f-values, thus providing smoother results and a larger approximation error, which is lower than 10^{-3}.

(a) Input res.: 50×50 (b) Interp.: 100×100 (c) LS: 100×100 (d) Approx. err.

Figure 6: (a) Input $f : \mathcal{P} \to \mathbb{R}$ and local approximation $F : \mathbb{R}^2 \to \mathbb{R}$ with RBFs and (b) interpolating or (c) LS constraints. (d) $\mathcal{L}_\infty$ error (y-axis) at each pixel (x-axis) of (c) with respect to a different k-nearest neighbor.



(a) $f : \mathcal{P} \to \mathbb{R}$ (b) $F : \mathbb{R}^3 \to \mathbb{R}$ (c) (d)

Figure 7: (a) Level-sets of a noisy map $f : \mathcal{P} \to \mathbb{R}$, achieved by perturbing with a Gaussian noise the samples of a smooth function $\tilde{f} : \mathcal{P} \to \mathbb{R}$ on the set $\mathcal{P}$ of vertices of a triangulated surface. (b) Iso-surfaces of the corresponding approximation. (c) Approximation error (y-axis) at each point (x-axis) of $\mathcal{P}$. (d) $\mathcal{L}_\infty$ error (y-axis) with respect to an increasing (x-axis) k-nearest neighbor used by the local LS approximation.

Choice of the approximation scheme and parameters. If the f-values are noise-free or have been smoothed, then the local interpolation (Fig. 5(a,c,e)) has the highest approximation accuracy. Otherwise (Fig. 5(b,d,f)), we compute the local approximation with LS constraints, which shows the best compromise between accuracy and robustness to noise. Since the number of points in each neighbor is generally small (e.g., $20 \le k \le 30$), we consider a generating map ϕ with a global support. Common choices of ϕ are the Gaussian $\phi(t) := \exp(-t/\sigma)$ and the bi-harmonic $\phi(t) := |t^3|/\sigma$ kernel, where the support σ is computed according to [33, 34] and adapted to the local sampling of $\mathcal{P}$.

Fig. 8 shows the local approximation of the electrostatic charge of the molecule C_6H_6. In Fig. 9, the local approximation with RBFs of medical data acquired on two consecutive slices of a human head has been resampled on an intermediate slice without performing a new acquisition, which is generally time-consuming and possibly intractable (e.g., nocivity, health problems, etc.).

2.2. Properties of the local approximation

To discuss the main features of the proposed approach, let us define the *discrete shape space*

$$\mathcal{F}(\mathcal{P}) := \{f : \mathcal{P} \to \mathbb{R},\ f \text{ scalar function on } \mathcal{P}\}. \quad (3)$$

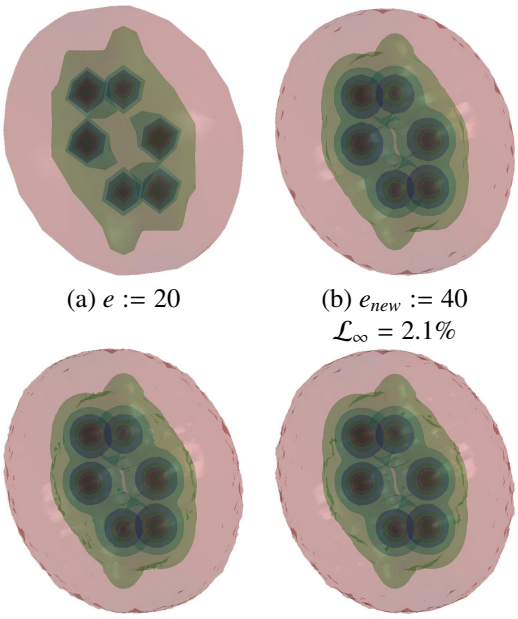
In $\mathcal{F}(\mathcal{P})$, any scalar function $f : \mathcal{P} \to \mathbb{R}$ is uniquely defined by the array $\mathbf{f} := (f(\mathbf{x}_i))_{i=1}^n$ of its values at $\mathcal{P}$ and $\mathcal{F}(\mathcal{P})$ is isomorphic to $\mathbb{R}^n$. We also introduce the *approximating operator*

$$\mathcal{A} : \mathcal{F}(\mathcal{P}) \to \mathcal{F}(\mathbb{R}^3), \qquad \mathbf{f} \mapsto \mathcal{A}(\mathbf{f}),$$

which maps the input function to its approximation.

Linearity. First of all, we show that the operator $\mathcal{A}$ is linear and we focus on the local approximation with LS constraints, which generalizes the interpolating case (i.e., $\mathcal{N}_\mathbf{x} = \overline{\mathcal{N}}_\mathbf{x}$). To this end, we verify that $F(\mathbf{x})$ can be written in the dual form

$$\begin{cases} F(\mathbf{x}) := \sum_{s=1}^h f(\mathbf{x}_{j_s})\rho_s(\mathbf{x}) = \rho^T(\mathbf{x})\mathbf{f}_h, \\ \rho(\mathbf{x}) := (\rho_s(\mathbf{x}))_{s=1}^h, \quad \mathbf{f}_h := (f(\mathbf{x}_{j_s}))_{s=1}^h, \end{cases} \qquad \mathbf{x} \in \mathbb{R}^3,$$



(a) $e := 20$ (b) $e_{new} := 40$, $\mathcal{L}_\infty = 2.1\%$ (c) $e_{new} := 100$, $\mathcal{L}_\infty = 2.4\%$ (d) $e_{new} := 200$, $\mathcal{L}_\infty = 2.6\%$

Figure 8: (a) Iso-surfaces of the electrostatic charge $E : \mathbb{R}^3 \to \mathbb{R}$ of the molecule C_6H_6 and sampled on a low resolution grid $\mathcal{P}$ ($e \times e \times e$). (b-d) Iso-surfaces of the local LS approximation $F : \mathbb{R}^3 \to \mathbb{R}$ of $\{E(\mathbf{x})\}_{\mathbf{x}\in\mathcal{P}}$ on a $e_{new} \times e_{new} \times e_{new}$ grid $\mathcal{P}^\star$. Here, $\mathcal{L}_\infty$ is the approximation error between F and E on $\mathcal{P}^\star$.

i.e., as a linear combination of the f-values and the dual functions $\{\rho_s(\mathbf{x})\}_{s=1}^h$. From the identity

$$\begin{aligned} F(\mathbf{x}) &= \sum_{s=1}^{k} \beta_s(\mathbf{x})\phi_{j_s}(\mathbf{x}) \\ &= \beta^T(\mathbf{x})\tilde{\phi}(\mathbf{x}), \quad \beta := (\beta_s)_{s=1}^k \quad \tilde{\phi}(\mathbf{x}) := (\phi_{j_s}(\mathbf{x}))_{s=1}^k, \\ &=_{(2)} \mathbf{f}_h^T \Phi(\Phi^T\Phi)^{-1}\tilde{\phi}(\mathbf{x}), \end{aligned}$$

we get the dual shape basis

$$\rho(\mathbf{x}) = \Phi(\Phi^T\Phi)^{-1}\tilde{\phi}(\mathbf{x}). \tag{4}$$

Given the functions $F(\mathbf{x}) := \rho^T(\mathbf{x})\mathbf{f}_h$, $G(\mathbf{x}) := \rho^T(\mathbf{x})\mathbf{g}_h$ underlying $f, g : \mathcal{P} \to \mathbb{R}$ in $\mathcal{N}_\mathbf{x}$, the linearity of $\mathcal{A}$ follows from the relation

$$\begin{aligned} \mathcal{A}(af + bg)(\mathbf{x}) &= \rho^T(\mathbf{x})(a\mathbf{f}_h + b\mathbf{g}_h) \\ &= aF(\mathbf{x}) + bG(\mathbf{x}) \\ &= a\mathcal{A}(f)(\mathbf{x}) + b\mathcal{A}(g)(\mathbf{x}) \quad a, b \in \mathbb{R}. \end{aligned}$$

Our main aim is to provide an accurate approximation $F : \mathbb{R}^3 \to \mathbb{R}$ of a discrete map $f : \mathcal{P} \to \mathbb{R}$. Indeed,

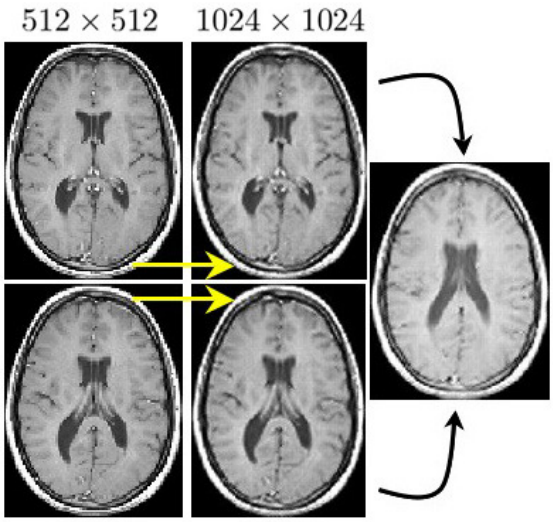


Figure 9: The input images (first column) have been approximated with the implicit LS approximation (second column) at a double resolution. Then, the resulting approximation has been sampled on a new slice in-between the input ones (right picture).

the output of the proposed approach is the discrete set $\{F(\mathbf{x})\}_{\mathbf{x}\in\mathcal{S}}$ of F-values at a set $\mathcal{S}$ of samples. Additional properties of the local approximation with RBFs, such as the continuity, are discussed in [35].

Computational cost. The computation of the nearest neighbor graph takes $O(n \log n)$ time [32]. Assuming that the neighbor $\mathcal{N}_\mathbf{x}$ contains k points, the evaluation of $F(\mathbf{x})$ requires to solve a $k \times k$ linear system, where k is generally small (i.e., $20 \le k \le 30$) and much lower than n. Its solution takes $O(k^3)$ time with direct solvers and varies from $O(k)$ to $O(k^2)$ in case of sparse coefficient matrix and iterative solvers [12]. Then, the evaluation of F at s sample points varies from $O(sk^3)$ to $O(sk)$. According to Table 1 and Fig. 10, the computational cost of the proposed approach is generally lower than the approximation with globally- and locally-supported RBFs. It also reduces the memory storage from $O(n^3)$ and $O(kn)$ to $O(k^3)$. Finally, our method has the same order of computational complexity of local approximation schemes, such as the MLS approximation [13, 14, 15] and the multi-level PU [18]. For the examples of the paper, Table 2 reports the minimum and maximum number of points in $\mathcal{N}_\mathbf{x}$, $\mathbf{x} \in \mathcal{P}$, which have been used for the evaluation of $F(\mathbf{x})$.

Stability and accuracy. To estimate the stability to noise of the local approximation scheme, let us perturb

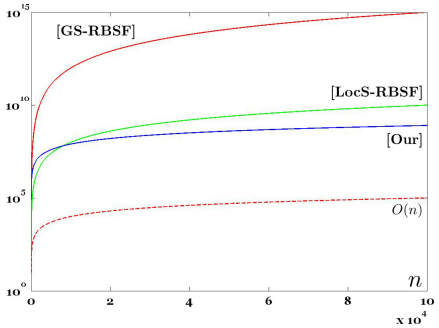


Figure 10: Computational cost (y-axis, log-scale) of different approximation schemes (Table. 1) with respect to the number n (x-axis) of input points.

the input f-values on $\mathcal{P}$ as $\{f(\mathbf{x}_i) + e_i\}_{i=1}^n$ and consider the corresponding approximations $F, F_{\mathbf{e}} : \mathbb{R}^3 \to \mathbb{R}$. The LS approximations $F(\mathbf{x}) = \beta^T(\mathbf{x})\tilde{\phi}(\mathbf{x})$ and $F_{\mathbf{e}}(\mathbf{x}) = \beta_{\mathbf{e}}^T(\mathbf{x})\tilde{\phi}(\mathbf{x})$ satisfy the normal equations

$$\begin{aligned} &(\Phi^T\Phi)\beta(\mathbf{x}) = \Phi^T\mathbf{f}_h, && \beta(\mathbf{x}) := (\beta_i(\mathbf{x}))_{i=1}^h, \\ &(\Phi^T\Phi)\beta_{\mathbf{e}}(\mathbf{x}) = \Phi^T(\mathbf{f}_h + \mathbf{e}_h), && \beta_{\mathbf{e}}(\mathbf{x}) := (\beta_{\mathbf{e}}^{(i)}(\mathbf{x}))_{i=1}^h, \end{aligned}$$

with $\mathbf{e}_h := (e_{j_s})_{s=1}^h$ perturbation vector. From these relations, the variation between $F(\mathbf{x})$ and $F_{\mathbf{e}}(\mathbf{x})$ in $\mathcal{N}_{\mathbf{x}}$ is bounded as

$$\begin{aligned} |F(\mathbf{x}) - F_{\mathbf{e}}(\mathbf{x})| &\leq_{(2)} \|\beta(\mathbf{x}) - \beta_{\mathbf{e}}(\mathbf{x})\|_2 \|\tilde{\phi}(\mathbf{x})\|_2 \\ &= k^{1/2}\|\phi\|_\infty \|(\Phi^T\Phi)^{-1}\Phi^T\mathbf{e}_h\|_2 \\ &\leq k^{1/2}\|\phi\|_\infty \frac{\|\Phi^T\|_2}{\lambda_1(\Phi^T\Phi)}\|\mathbf{e}\|_2. \end{aligned} \tag{5}$$

Then, the stability to noise of the proposed approach is mainly controlled by the inverse of the minimum eigenvalue of the coefficient matrix $\Phi^T\Phi$ and the noise magnitude $\|\mathbf{e}\|_2$. The minimum eigenvalue is efficiently computed using iterative methods for the evaluation of the matrix spectrum [12]. The point-wise approximation error between the input function value $f(\mathbf{x}_i) = \mathbf{e}_i^T\mathbf{f}$ and its approximation $F(\mathbf{x}_i)$ is estimated as

$$\begin{aligned} |F(\mathbf{x}_i) - f(\mathbf{x}_i)| &\leq \|\rho(\mathbf{x}_i) - \mathbf{e}_i\|_2 \|\mathbf{f}_h\|_2 \\ &=_{(4)} \|\Phi(\Phi^T\Phi)^{-1}\tilde{\phi}(\mathbf{x}_i) - \mathbf{e}_i\|_2 \|\mathbf{f}_h\|_2 \\ &\leq \left(1 + \frac{\|\Phi\|_2}{\lambda_1(\Phi^T\Phi)}\right)\|\mathbf{f}\|_2. \end{aligned} \tag{6}$$

If $\Phi^T\Phi$ is ill-conditioned, then its preconditioning [12] (Ch. 10) improves the approximation accuracy (5) and computation stability (6).

2.3. Applications and discussion

We now compare the local approximation with RBFs to previous work on image and surface approximation.

Table 1: Computational cost of different approximations (Fig. 10): solution of the normal equation (*LSys.*); evaluation of F at $\mathbf{x}$ (*Ev.* $\mathbf{x}$) and a set $\mathcal{S}$ of s samples (*Ev.* $\mathcal{S}$). The column (*Con.*) indicates the type of constraints; i.e., interpolating (*In.*) and least-squares (*LS.*).

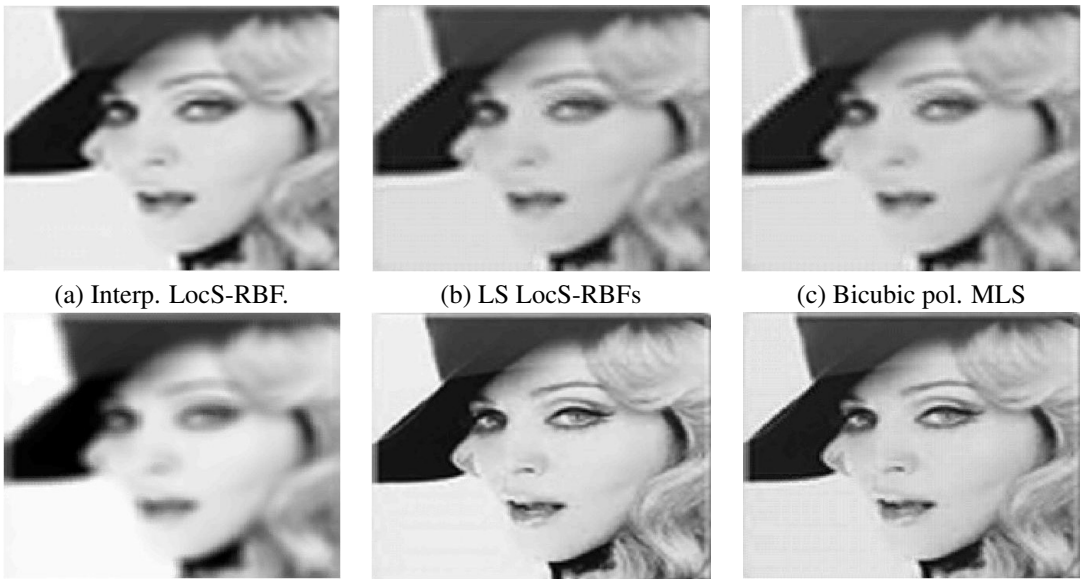
Approx. Scheme	**LSys.**	**Ev. x**	**Ev. $\mathcal{S}$**	**Con.**
Our method	$O(k^3)$	$O(k)$	$O(sk^3)$	In./Ls.
MLS [13, 14, 15]	$O(k^3)$	$O(k)$	$O(sk^3)$	Ls.
PU [18]	$O(k^3)$	$O(k)$	$O(sk^3)$	Ls.
LocS-RBF [36]	$O(n^2)$	$O(n)$	$O(sk^3)$	In./Ls.
GS-RBF [17]	$O(n^3)$	$O(n)$	$O(sk^3)$	In./Ls.

Table 2: Minimum and maximum number of points in $\mathcal{N}_{\mathbf{x}}$, $\mathbf{x} \in \mathcal{P}$, used for the computation of $F(\mathbf{x})$.

Test	$n := \lvert\mathcal{P}\rvert$	**Min**	**Max**
Fig. 1	$262K$	12	27
Fig. 7	$8K$	10	18
Fig. 8	$64K$	12	27
Fig. 16	$8K$	14	29
Fig. 17(a)	$11K$	10	25
Fig. 17(b)	$6K$	12	14

Image resampling. Computing different approximations (Fig. 11, Table 3) of the input data in Fig. 6(a) and resampling them on a higher resolution grid, we show the good extrapolation properties of the proposed approach (Fig. 11(e,f)) with respect to (i) the approximation with locally-supported RBFs and interpolating (Fig. 11(a)) or LS (Fig. 11(b)) constraints; (ii) the polynomial MLS (Fig. 11(c)); and (iii) the PU (Fig. 11(d)). Furthermore, the local approximations with RBFs and interpolating (Fig. 11(e)) or LS (Fig. 11(f)) constraints show the preservation of local details and a reduction of the blurring effects in Fig. 11(a-d).

According to Tables 3, 4, the extrapolation capability of the approximation with RBFs is higher than polynomial approximations. This property is due to the use of RBFs, which are adapted to the local shape better than polynomials, and the LS approach, which smoothes the pixels by minimizing the LS error. Indeed, the proposed approach is better suited to extrapolate local information as it happens when upsampling low-resolution images. For instance, in Fig. 12 the blurred effects in the approximations disappear when the input resolution is 300×300 or higher. The redundant information available in high-resolution images (Fig. 13) is effectively removed with linear downsampling methods, without applying accurate schemes such as the local approxi-



(a) Interp. LocS-RBF. (b) LS LocS-RBFs (c) Bicubic pol. MLS

(d) PU (e) Local interp. with RBFs (f) Local LS with RBFs

Figure 11: 200×200 resamplings of the image in Fig. 6(a) with compactly-supported RBFs and (a) interpolating or (b) LS constraints. (c) Polynomial MLS and (d) PU approximation. The local approximations with RBFs with (e) interpolating and (f) LS constraints show the preservation of local details and a reduction of the blurring effects in (a-d). Statistics are reported in Table 3.

mation with RBFs.

Implicit surface reconstruction. In $\mathbb{R}^3$, implicit modeling [37] and MLS techniques provide an easy way to approximate a point set with an iso-surface. The quality of the input data $\mathcal{P}$ in terms of sampling and noise is usually represented by a *confidence map* $f : \mathcal{P} \to [0, 1]$, which associates to each point $\mathbf{x}_i$ of $\mathcal{P}$ its degree $f(\mathbf{x}_i)$ of reliability. This map is provided by the acquisition process, as it happens for laser scanners, or computed by analyzing the data variability through likelihood estimation [38]. A higher value of f corresponds to a higher reliability. Assuming that the confidence map f is known, we approximate the data set $\{(\mathbf{x}_i, f(\mathbf{x}_i))\}_{i=1}^n$ with the iso-surface $\Sigma := \{\mathbf{x} \in \mathbb{R}^3 : F(\mathbf{x}) = 1\}$, where $F : \mathbb{R}^3 \to \mathbb{R}$ is the function underlying the confidence map f (Fig. 14). Furthermore, interpolating constraints preserve the sharp features of the underlying shape and LS constraints guarantee the stability of the local LS approximation with respect to irregular sampling and noise (Fig. 15). Note that both types of approximations can be combined and used to achieve a different approximation accuracy and smoothness.

Applying interpolating constraints at those points of $\mathcal{P}$ whose confidence value is one guarantees that these points will belong to the iso-surface Σ. Surface irregularities might appear in regions with a low sampling density or when the evaluation point $\mathbf{x}$ is chosen far from $\mathcal{P}$; in this case, it is enough to enlarge the neighbor radius or use the k-nearest neighbor of $\mathbf{x}$. The main difference between the σ- or k-nearest neighbor graph is the support of the resulting approximation $F : \mathbb{R}^3 \to \mathbb{R}$; the former provides a globally-supported approximation. In the latter case, the approximation generally has a local support, which is localized around $\mathcal{P}$.

Fig. 16 shows the reconstruction obtained by the lo-

Table 3: Timings (*TT*, s:ms), LS (*LS ex. er.*) and bicubic MLS (*MLS ex. er.*) extrapolation error associated to the image upsampling from a $n \times n$ to a 200×200 resolution (Fig. 11). The extrapolation error has been computed as the $\mathcal{L}_\infty$ error between each resampling and the ground-truth 200×200 image. Tests performed on a 2.7 GHz Intel Core i7, with 8GB memory.

n	**TT**	**LS ex. er.**	**TT**	**MLS ex. er.**
25	5.18	$1.66 \cdot 10^{-1}$	5.01	$2.25 \cdot 10^{-1}$
50	8.40	$0.72 \cdot 10^{-1}$	8.45	$1.04 \cdot 10^{-1}$
100	10.27	$7.68 \cdot 10^{-2}$	11.02	$9.61 \cdot 10^{-2}$
150	12.85	$1.71 \cdot 10^{-2}$	13.03	$3.06 \cdot 10^{-2}$
200	14.09	$1.12 \cdot 10^{-3}$	14.00	$2.24 \cdot 10^{-3}$

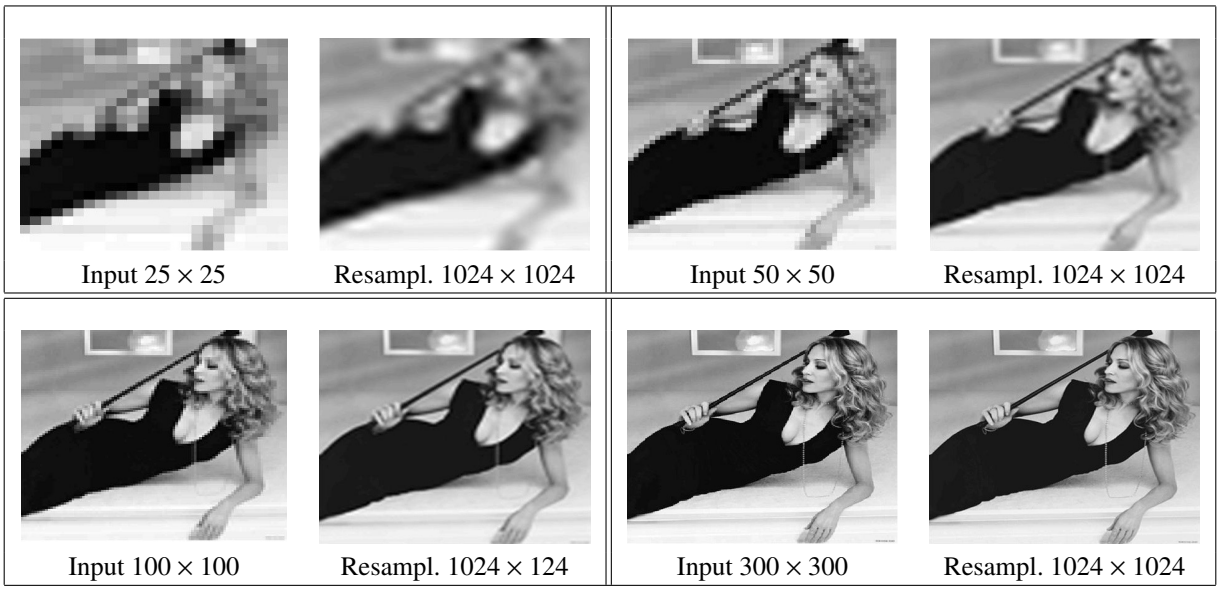


Figure 12: Images at different resolutions and 1024×1024 local LS resamplings. Statistics are reported in Table 4.

(a) Input 1024×1024 (b) 200×200 Local LS (c) 200×200 Bicubic MLS

Figure 13: (a) Input and downsampled image with the (b) local LS and (c) bicubic MLS approximation.

Table 4: $\mathcal{L}_\infty$ extrapolation error related to the local LS approximation with RBFs (*LS ex. er.*) and the MLS approximation with bicubic polynomials (*MLS ex. er.*) of an input $n \times n$ image (Fig. 12).

n	**LS ex. er.**	**MLS ex. er.**
25	$1.26 \cdot 10^{-1}$	$2.10 \cdot 10^{-1}$
50	$1.35 \cdot 10^{-2}$	$8.52 \cdot 10^{-2}$
100	$3.94 \cdot 10^{-2}$	$5.90 \cdot 10^{-2}$
300	$1.66 \cdot 10^{-2}$	$2.69 \cdot 10^{-2}$
500	$1.27 \cdot 10^{-2}$	$1.61 \cdot 10^{-2}$
700	$6.43 \cdot 10^{-3}$	$9.85 \cdot 10^{-3}$

cal LS approximation with RBFs; the *algebraic point set surface* [39, 40], which is based on the local fitting of algebraic surfaces; and the *robust implicit MLS* method [41], which preserves sharp features using linear regression techniques. From this example, we get that our method is useful for approximating point sets that are noisy or have been sampled at low resolution. In fact, the local LS approximation with RBFs reduces the number of local artifacts, which are typically introduced where a coarse discretization of the normals is used to recover the interior and exterior part of the reconstructed surface and to avoid the constant trivial solution [36, 42, 17]. For densely sampled shapes, polynomial MLS surfaces already provide satisfactory results in terms of feature preservation, accuracy, and computational cost. Indeed, the local LS approximation with RBFs improves the approximation accuracy at the price of a higher computational cost.

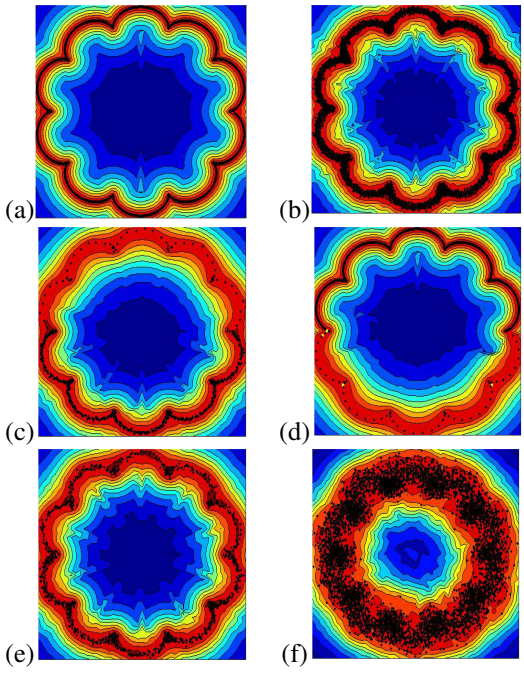


Figure 14: Level-sets of the approximation $F : \mathbb{R}^2 \to \mathbb{R}$ underlying the confidence map $f : \mathcal{P} \to [0, 1]$ on $\mathcal{P}$ (black dots), which is (a) regularly-, (b-d) irregularly-sampled, (e,f) noisy.

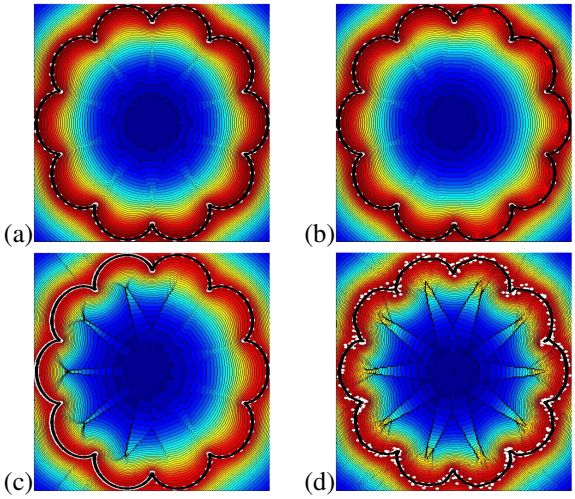


Figure 15: Level-sets (black curves) and reconstruction (bold curve) $\Sigma := F^{-1}(-1)$ of the local approximation F of the input data (white dots) with (a-c) different densities and (d) noise. Curve spikes are recovered with interpolating constraints.

3. Volumetric approximation through the Laplacian spectrum

We now apply the local approximation with RBFs to scalar functions defined on triangulated surfaces. In [21], the approximation of Laplacian eigenvectors on point sets, whose accuracy is limited to the close proximity of the underlying surface, is computed through the Nystrom method. For the *spectral volumetric approximation*, the coefficients' vector β in Eq. (1) is independent of the evaluation point $\mathbf{x}$ and is achieved by projecting the array $\mathbf{f} := (f(\mathbf{x}_i))_{i=1}^n$ of the values of the input discrete signal $f : \mathcal{P} \to \mathbb{R}$ along the eigenvectors of the Laplacian matrix associated to the input data set $\mathcal{P}$. These basis functions, which are the Laplacian eigenvectors associated to the input shape, take into account its local geometry. With respect to the local approximation with interpolating and LS constraints, the most time-consuming part of the spectral volumetric approximation is the computation of the Laplacian spectrum in super-linear time [43].

If $\mathcal{P}$ is the set of vertices of a triangle mesh $\mathcal{M}$, then the discrete shape space $\mathcal{F}(\mathcal{P})$ in Eq. (3), is endowed with the scalar product $\langle \mathbf{f}, \mathbf{g} \rangle_B := \mathbf{f}^T B \mathbf{g}$, $\mathbf{f}, \mathbf{g} \in \mathbb{R}^n$, induced by a $n \times n$ positive-definite matrix B. Examples of B are the identity matrix, which induces the Euclidean scalar product, and the mass matrix of the linear FEM discretization [44] of the Laplace-Beltrami operator. More precisely, the linear FEM discretization [43, 44] is given by the *weighted Laplacian matrix* $L := B^{-1}\tilde{L}$. Here, the *stiffness matrix* $\tilde{L}$ is the un-normalized Tutte-Laplacian matrix with cotangent weights and the *mass matrix* B codes the geometry of $\mathcal{M}$ in terms of triangle areas. These matrices are defined as

$$B_{ij} := \begin{cases} \frac{|t_r|+|t_s|}{12} & j \in N(i), \\ \frac{\sum_{k \in N(i)} |t_k|}{6} & i = j, \\ 0 & \text{else}, \end{cases}$$

$$\tilde{L}_{ij} := \begin{cases} w_{ij} := \frac{\cot \alpha_{ij} + \cot \beta_{ij}}{2} & j \in N(i), \\ -\sum_{k \in N(i)} w_{ik} & i = j, \\ 0 & \text{else}, \end{cases}$$

where $N(i)$ is the 1-star of the vertex i; α_{ij}, β_{ij} are the angles opposite to the edge (i, j); t_r, t_s are the triangles that share the edge (i, j); and $|t|$ is the area of the triangle t. Then, the corresponding eigensystem $\{(\lambda_i, \mathbf{v}_i)\}_{i=1}^n$ satisfies the generalized eigenproblem

$$\tilde{L}\mathbf{v}_i = \lambda_i B \mathbf{v}_i, \quad \langle \mathbf{v}_i, \mathbf{v}_j \rangle_B = \mathbf{v}_i^T B \mathbf{v}_j = \delta_{ij}, \quad i, j = 1, \ldots, n.$$

In this case, the function $F : \mathbb{R}^3 \to \mathbb{R}$ underlying $f : \mathcal{P} \to \mathbb{R}$ is computed through the following steps:

- the array $\mathbf{f}$ of f-values is expressed as a linear combination of the Laplacian eigenvectors $\mathcal{V} := \{\mathbf{v}_i\}_{i=1}^n$; i.e., $\mathbf{f} = \sum_{i=1}^n \langle \mathbf{f}, \mathbf{v}_i \rangle_B \mathbf{v}_i$;

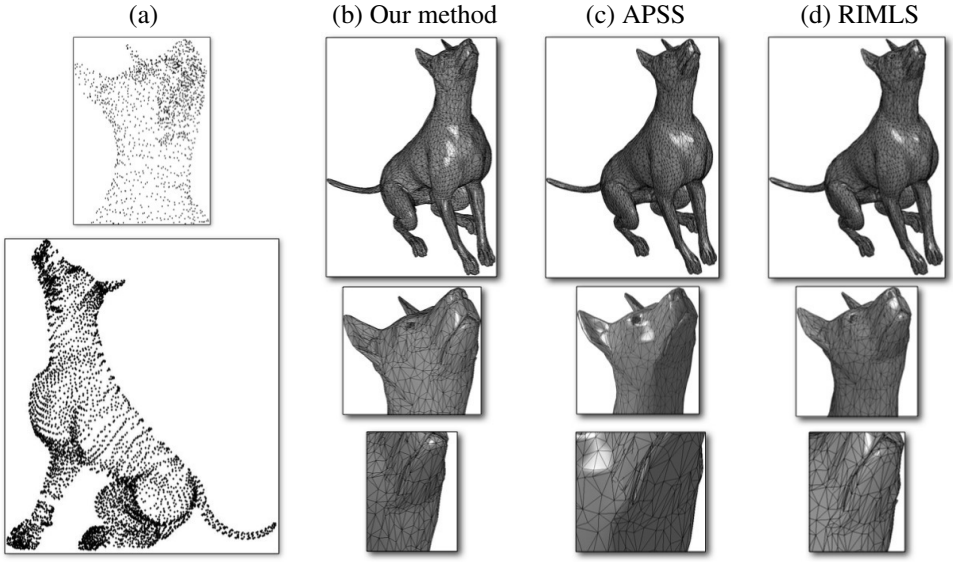


Figure 16: Reconstruction of (a) a coarse point set with the (b) local approximation with RBFs, (c) *algebraic point set surface* (APSS), and (d) *robust implicit MLS* (RIMLS). (c,d-bottom) Non-manifold areas appear on the mouth.

- using the local approximation described in Section 2.1, each eigenvector $\mathbf{v}_i \in \mathcal{F}(\mathcal{P})$ is extended to a volumetric map $\phi_i : \mathbb{R}^3 \to \mathbb{R}$;
- the maps $\{\phi_i\}_{i=1}^n$ and the coefficient vector $\beta(\mathbf{x}) := \mathbf{f}^T BV$, $V = [\mathbf{v}_1, \dots, \mathbf{v}_n]$, are used to recover the function

$$F(\mathbf{x}) := \sum_{i=1}^{n} \langle \mathbf{f}, \mathbf{v}_i \rangle_B \phi_i(\mathbf{x}) = \mathbf{f}^T BV\tilde{\phi}(\mathbf{x}), \qquad (7)$$

  as a linear combination of the eigenfunctions $\{\phi_i\}_{i=1}^n$ associated to the Laplacian eigenvectors $\{\mathbf{v}_i\}_{i=1}^n$, whose coefficients are those that express $\mathbf{f}$ in terms of the Laplacian eigenvectors.

Then, the *approximating functional* is defined as $\mathcal{A}(f)(\mathbf{x}) = \sum_{i=1}^n \langle \mathbf{f}, \mathbf{v}_i \rangle_B \phi_i(\mathbf{x})$, $\mathbf{x} \in \mathbb{R}^3$.

An example of volumetric approximation of a Laplacian eigenfunction is shown in Fig. 17. Since we have applied the local approximation with interpolating constraints, the iso-surface $F^{-1}(\alpha)$ intersect the surface $\mathcal{M}$ along the level-set $f^{-1}(\alpha)$.

Proposition 3.1. *The spectral volumetric approximation satisfies the following properties:*

(A) it is linear with respect to the input f-values;

(B) the approximation error between F and f on $\mathcal{P}$ is bounded as follows

$$\|(F(\mathbf{x}_s))_{s=1}^n - \mathbf{f}\|_2 \leq [\lambda_n(\Phi) + \lambda_n(V)]\, \lambda_n(V)\lambda_n(B)\|\mathbf{f}\|_2;$$

(C) perturbing the input data $\mathbf{f} := (f(\mathbf{x}_i))_{i=1}^n$ with a noise vector $\mathbf{e} := (e_i)_{i=1}^n$, the discrepancy between the corresponding approximations is bounded as

$$\|\mathcal{A}(\mathbf{f} + \mathbf{e}) - \mathcal{A}(\mathbf{e})\|_\infty \leq \|\phi\|_\infty \|\mathbf{e}\|_B.$$

Proof. (A) The linearity of the approximation follows from the identity

$$\begin{aligned}\mathcal{A}(a\mathbf{f} + b\mathbf{g}) &= \sum_{i=1}^{n} \langle a\mathbf{f} + b\mathbf{g}, \mathbf{v}_i \rangle_B \phi_i \\ &= a \sum_{i=1}^{n} \langle \mathbf{f}, \mathbf{v}_i \rangle_B \phi_i + b \sum_{i=1}^{n} \langle \mathbf{g}, \mathbf{v}_i \rangle_B \phi_i \\ &= a\mathcal{A}(\mathbf{f}) + b\mathcal{A}(\mathbf{g}), \quad a, b \in \mathbb{R}, \quad \mathbf{f}, \mathbf{g} \in \mathcal{F}(\mathcal{P}).\end{aligned}$$

(B) We derive the approximation error between f and F on $\mathcal{P}$. From Eq. (7), we get that

$$(F(\mathbf{x}_s))_{s=1}^n = (\tilde{\phi}(\mathbf{x}_s))_{s=1}^n \mathbf{a} = \Phi\mathbf{a}, \quad \Phi := (\phi_i(\mathbf{x}_j))_{i,j=1}^n,$$

where $\mathbf{a} := (\langle \mathbf{f}, \mathbf{v}_i \rangle_B)_{i=1}^n \in \mathbb{R}^n$ is the array of the components of f along the Laplacian eigenfunctions (i.e., $\mathbf{f} := V\mathbf{a}$). Indeed, the approximation error between F and f on $\mathcal{P}$ is bounded as follows

$$\begin{aligned}\|(F(\mathbf{x}_s))_{s=1}^n - \mathbf{f}\|_2 &= \|(\Phi - V)\mathbf{a}\|_2 \\ &\leq [\lambda_n(\Phi) + \lambda_n(V)]\, \lambda_n(B)\|V\|_2\|\mathbf{f}\|_2.\end{aligned}$$

(C) The discrepancy between $\mathcal{A}(\mathbf{f} + \mathbf{e})$ and $\mathcal{A}(\mathbf{f})$ is

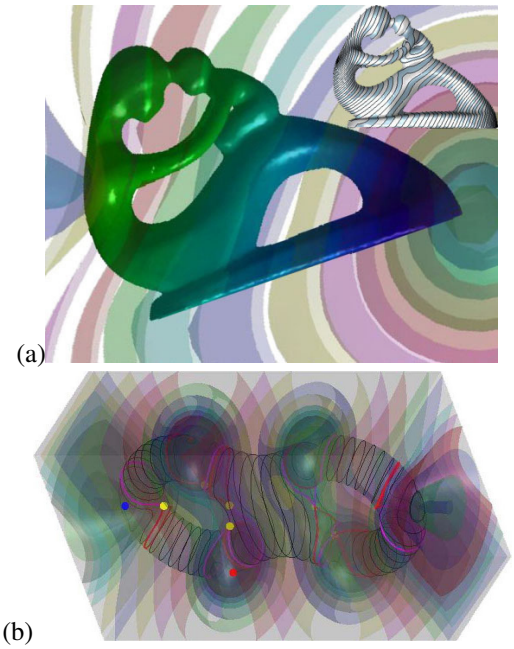


Figure 17: (a,b) Level-sets of a Laplacian eigenfunction $f : \mathcal{P} \to \mathbb{R}$ on a surface mesh and iso-surfaces of the corresponding local approximations $F : \mathbb{R}^3 \to \mathbb{R}$ with interpolating constraints.

bounded as

$$\begin{aligned} \|\mathcal{A}(\mathbf{f}+\mathbf{e}) - \mathcal{A}(\mathbf{e})\|_\infty &= \|\langle \mathbf{e}, \mathbf{v}_i \rangle_B \phi_i\|_\infty \\ &\leq \sum_{i=1}^{n} |\langle \mathbf{e}, \mathbf{v}_i \rangle_B| \|\phi_i\|_\infty \\ &\leq \|\phi\|_\infty \|\mathbf{e}\|_B. \end{aligned}$$

4. Conclusions and future work

This paper has presented a meshless approximation of discrete scalar functions defined on 3D shapes and volumetric domains. The approximation scheme combines interpolating and least-squares conditions; shows a high approximation accuracy and extrapolation capabilities; is computationally efficient through the solution of a small and well-conditioned linear system.

Encoding the input discrete map into an implicit representation also provides an efficient way to accurately compute high-level representations such as the Reeb graph [45] and the Morse complex [46]; apply physically-based modeling [4, 47, 48]; and classify the critical points of implicit maps defined as linear combinations of RBFs [49, 50]. Finally, a volume-based approximation of a surface-based map gives an insight and could be useful to make predictions about the underlying phenomenon. For instance, the volumetric approximation of spatio-physico-chemical properties measured or simulated on a molecule could be used to predict interactions [51].

Acknowledgements

Special thanks are given to the anonymous reviewers for their valuable comments, which helped us to improve the content of the paper. Images and shapes are courtesy of `www.madonna.com` and the AIM@SHAPE Repository. This work has been partially supported by the FP7 Marie Curie Initial Training Network *Multi-ScaleHuman*.

References

[1] V. Jain, H. Zhang, A spectral approach to shape-based retrieval of articulated 3D models, Computer Aided Design 39 (5) (2007) 398–407.

[2] X. Li, X. Gu, H. Qin, Surface mapping using consistent pants decomposition, IEEE Transactions on Visualization and Computer Graphics 15 (4) (2009) 558–571.

[3] X. Li, H. Xu, S. Wan, Z. Yin, W. Yu, Feature-aligned harmonic volumetric mapping using MFS, Computers & Graphics (2010) 242–251.

[4] T. Martin, E. Cohen, M. Kirby, Volumetric parameterization and trivariate b-spline fitting using harmonic functions, in: Proc. of the ACM Symposium on Solid and Physical Modeling, 2008, pp. 269–280.

[5] T. Martin, E. Cohen, Volumetric parameterization of complex objects by respecting multiple materials, Computer Graphics 34 (2010) 187–197.

[6] R. M. Rustamov, Interpolated eigenfunctions for volumetric shape processing, The Visual Computer 27 (11) (2011) 951–961.

[7] R. Rustamov, Template-based shape descriptor, Eurographics Workshop on 3D Object Retrieval (2009) 1–7.

[8] R. M. Rustamov, Multiscale biharmonic kernels, Computer Graphics Forum 30 (5) (2011) 1521–1531.

[9] R. M. Rustamov, Laplace-Beltrami eigenfunctions for deformation invariant shape representation, in: Proc. of the Symposium on Geometry processing, 2007, pp. 225–233.

[10] Y. Lipman, R. Rustamov, T. Funkhouser, Biharmonic distance, ACM Transactions on Graphics 29 (3).

[11] J. Sun, M. Ovsjanikov, L. J. Guibas, A concise and provably informative multi-scale signature based on heat diffusion, Computer Graphics Forum 28 (5) (2009) 1383–1392.

[12] G. Golub, G. VanLoan, Matrix Computations, John Hopkins University Press, 2nd Edition, 1989.

[13] N. Dyn, D. Levin, S. Rippa, Numerical procedures for surface fitting of scattered data by radial functions, SIAM Journal on Scientific and Statistical Computing 7(2) (1986) 639–659.

[14] C. A. Micchelli, Interpolation of scattered data: Distance matrices and conditionally positive definite functions, Constructive Approximation 2 (1986) 11–22.

[15] H. Wendland, Real piecewise polynomial, positive definite and compactly supported radial functions of minimal degree, Advances in Computational Mathematics 4(4) (1995) 389–396.

[16] S. Martin, P. Kaufmann, M. Botsch, M. Wicke, M. Gross, Polyhedral finite elements using harmonic basis functions, Computer Graphics Forum 27 (5) (2008) 1521–1529.
[17] G. Turk, J. F. O'Brien, Modelling with implicit surfaces that interpolate, ACM Siggraph 21 (4) (2002) 855–873.
[18] Y. Ohtake, A. Belyaev, M. Alexa, G. Turk, H.-P. Seidel, Multi-level partition of unity implicits, ACM Siggraph 22 (3) (2003) 463–470.
[19] J. Jin, M. Garland, E. A. Ramos, MLS-based scalar fields over triangle meshes and their application in mesh processing, in: Proc. of the Symposium on Interactive 3D Graphics and Games, 2009, pp. 145–153.
[20] G. Patanè, M. Spagnuolo, B. Falcidieno, Topology- and error-driven extension of scalar functions from surfaces to volumes, ACM Transactions on Graphics 29 (1) (2009) 1–20.
[21] R. R. Coifman, S. Lafon, Diffusion maps, Applied and Computational Harmonic Analysis 21 (1) (2006) 5–30.
[22] C. S. Co, B. Heckel, H. Hagen, B. Hamann, K. Joy, Hierarchical clustering for unstructured volumetric scalar fields, in: IEEE Visualization, 2003, p. 43.
[23] Y. Jang, M. Weiler, M. Hopf, J. Huang, D. S. Ebert, K. P. Gaither, T. Ertl, Interactively visualizing procedurally encoded scalar fields, in: Proc. of Symposium on Visualization, 2004, pp. 35–44.
[24] M. Weiler, R. Botchen, S. Stegmaier, T. Ertl, J. Huang, Y. Jang, D. Ebert, K. Gaither, Hardware-assisted feature analysis and visualization of procedurally encoded multifield volumetric data, Computer Graphics Applications 25 (5) (2005) 72–81.
[25] Y. Jang, R. P. Botchen, A. Lauser, D. S. Ebert, K. P. Gaither, T. Ertl, Enhancing the interactive visualization of procedurally encoded multifield data with ellipsoidal basis functions., Computer Graphics Forum 25 (3) (2006) 587–596.
[26] W. Hong, N. Neopytou, A. Kaufman, Constructing 3D elliptical gaussian for irregular data, in: Mathematical Foundations of Scientific Visualization, Computer Graphics, and Massive Data Visualization, 2006, pp. 213–225.
[27] K. Madsen, H. B. Nielsen, O. Tingleff, Methods for Non-Linear Least Squares Problems, 2nd Edition, Informatics and Mathematical Modelling, Technical University of Denmark, DTU, 2004.
[28] G. Patanè, B. Falcidieno, Computing smooth approximations of scalar functions with constraints, Computer & Graphics 33 (3) (2009) 399–413.
[29] X. Li, X. Guo, H. Wang, Y. He, X. Gu, H. Qin, Harmonic volumetric mapping for solid modeling applications, in: Proc. of Symposium on Solid and Physical Modeling, 2007, pp. 109–120.
[30] H. Wang, M. Jin, Y. He, X. Gu, H. Qin, User-controllable polycube map for manifold spline construction, in: Symposium on Solid and Physical Modeling, 2008, pp. 397–404.
[31] M. Pauly, R. Keiser, L. P. Kobbelt, M. Gross, Shape modeling with point-sampled geometry, ACM Transactions on Graphics 22 (3) (2003) 641–650.
[32] S. Arya, D. M. Mount, N. S. Netanyahu, R. Silverman, A. Y. Wu, An optimal algorithm for approximate nearest neighbor searching fixed dimensions, Journal of the ACM 45 (6) (1998) 891–923.
[33] T. K. Dey, J. Sun, An adaptive MLS surface for reconstruction with guarantees, in: ACM Symposium on Geometry Processing, 2005, pp. 43–52.
[34] N. J. Mitra, A. Nguyen, Estimating surface normals in noisy point cloud data, in: Proc. of Computational Geometry, ACM Press, 2003, pp. 322–328.
[35] G. Patanè, Moving least-squares approximation of d-dimensional data with radial basis functions, CNR-IMATI Technical Report 10.
[36] B. S. Morse, T. S. Yoo, D. T. Chen, P. Rheingans, K. R. Subramanian, Interpolating implicit surfaces from scattered surface data using compactly supported radial basis functions, in: IEEE Shape Modeling and Applications, 2001, pp. 89–98.
[37] J. Bloomenthal, B. Wyvill (Eds.), Introduction to Implicit Surfaces, Morgan Kaufmann Publishers Inc., 1997.
[38] M. Pauly, N. Mitra, L. Guibas, Uncertainty and variability in point cloud surface data, Proc. of the Symposium on Point-based Graphics (2004) 77–84.
[39] G. Guennebaud, M. Gross, Algebraic point set surfaces, ACM Transactions on Graphics 26.
[40] G. Guennebaud, M. Germann, M. H. Gross, Dynamic sampling and rendering of algebraic point set surfaces, Computer Graphics Forum 27 (2) (2008) 653–662.
[41] A. C. Öztireli, G. Guennebaud, M. H. Gross, Feature preserving point set surfaces based on non-linear kernel regression, Computer Graphics Forum 28 (2) (2009) 493–501.
[42] C. Shen, J. F. O'Brien, J. R. Shewchuk, Interpolating and approximating implicit surfaces from polygon soup, in: ACM Siggraph Courses, 2005, p. 204.
[43] B. Vallet, B. Levy, Manifold harmonics, Computer Graphics Forum 27(2).
[44] M. Reuter, F.-E. Wolter, N. Peinecke, Laplace-Beltrami spectra as Shape-DNA of surfaces and solids, Computer-Aided Design 38 (4) (2006) 342–366.
[45] V. Pascucci, G. Scorzelli, P. Bremer, A. Mascarenhas, Robust on-line computation of Reeb graphs: simplicity and speed, ACM Transactions on Graphics 26 (3).
[46] A. Gyulassy, V. Natarajan, V. Pascucci, B. Hamann, Efficient computation of Morse-Smale complexes for three-dimensional scalar functions, IEEE Transactions on Visualization and Computer Graphics 13 (6) (2007) 1440–1447.
[47] B. Adams, M. Wicke, Meshless approximation methods and applications in physics based modeling and animation, in: Eurographics Tutorials, 2009, pp. 213–239.
[48] B. Adams, M. Wicke, M. Ovsjanikov, M. Wand, H.-P. Seidel, L. Guibas, Meshless shape and motion design for multiple deformable objects, Computer Graphics Forum 29 (1) (2010) 43–59.
[49] J. C. Hart, A. Durr, D. Harsh, Critical points of polynomial metaballs, in: Proc. of Implicit Surfaces, 1998, pp. 69–76.
[50] S.-T. Wu, M. De Gomensoro, On improving the search for critical points of implicit functions, Proc. Implicit Surfaces (1999) 73–80.
[51] G. Cipriano, M. Gleicher, Molecular surface abstraction, IEEE Transactions on Visualization and Computer Graphics 13 (6) (2007) 1608–1615.