Surface inpainting with sparsity constraints ☆

Ming Zhong *, Hong Qin

Department of Computer Science, Stony Brook University, Stony Brook, NY, USA

A B S T R A C T

In this paper we devise a new algorithm for completing surface with missing geometry and topology founded upon the theory and techniques of sparse signal recovery. The key intuition is that any meaningful 3D shape, represented as a discrete mesh, almost certainly possesses a low-dimensional intrinsic structure, which can be expressed as a sparse representation in some transformed domains. Instead of estimating the missing geometry directly, our novel method is to find this low-dimensional representation which describes the entire original shape. More specifically, we find that, for many shapes, the vertex coordinate function can be well approximated by a very sparse coefficient representation with respect to the dictionary comprising its Laplacian eigenfunctions, and it is then possible to recover this sparse representation from partial measurements of the original shape. Taking advantage of the sparsity cue, we advocate a novel variational approach for surface inpainting, integrating data fidelity constraints on the shape domain with coefficient sparsity constraints on the transformed domain. Because of the powerful properties of Laplacian eigenbasis, the inpainting results of our method tend to be smooth and globally coherent with the remaining shape. We demonstrate the performance of our new method via various examples in geometry restoration, shape repair, and hole filling.

© 2015 Elsevier B.V. All rights reserved.

1. Introduction

In principle, surface inpainting refers to the completion or recovery of missing shape geometry based on the shape information that is currently available. The most prominent application of surface inpainting is mesh repair. Due to factors such as occlusions, low reflectance, and quality limitations of scanning equipments, 3D models generated from range scanners often contain holes that need to be filled; sometimes the source model itself has missing pieces and requires digital repair to attain a complete model. Another common application of surface inpainting is to remove geometric features because of shape editing needs. This is achieved by replacing the unwanted shape regions with inpainting patches.

From the statistical point of view, surface inpainting can be viewed as an estimation problem which infers the missing geometry from the observable shape, and the inpainting result is determined by the statistical model we have adopted. Generally speaking, there is no universally acceptable "correct" estimation; selecting the best inpainting is usually subjective or dependent on the requirement of downstream applications.

Most existing surface inpainting methods tackle the problem only in the mesh domain. These methods typically employ some geometric constraints as heuristics to ensure that the obtained inpainting patch is visually pleasing and blends

naturally with its neighboring geometry. The primary issue of geometry-constrained inpainting is that these methods only utilize the shape information in the vicinity of missing regions rather than consider the model in question in its entirety.

Our new surface inpainting method documented in this paper is inspired by the theory of sparse signal recovery and compressed sensing. The intuition is that for a meaningful 3D model, even though its global geometry is a high-dimensional signal, it most likely has a low-dimensional intrinsic structure. That is to say, the high-dimensional shape signal actually lives in a low-dimensional subspace, which can be captured by a sparse coefficient representation in some transformed (e.g., Fourier) domains. In another word, the coordinate function of a shape with N vertices can be decomposed as or well approximated by the linear combination of $k \ll N$ basis signals. According to the compressed sensing theory, the sparse coefficient representation can be recovered from partial measurements as long as the signal is sufficiently sparse and the sensing matrix satisfies certain properties (Candes et al., 2006).

The critical idea of our new inpainting algorithm is to estimate the spectral coefficient representation of the shape geometry from partial observations by imposing sparsity constraints on the reconstructed coefficients, exploiting the fact that most 3D models are highly compressible with respect to their Laplacian eigenfunctions (Karni and Gotsman, 2000). To the best of our knowledge, the utility of Laplacian eigenbasis towards the shape inpainting application has not yet been explored in the past. The estimation problem can be formulated with a data term strongly emphasizing fidelity to the observations and a penalty term constraining sparsity of the representation. Thus, the surface inpainting could be transformed to a sparse signal recovery problem and can be solved by either l_0 or l_1 optimization techniques. Such effort represents our first attempt towards technical innovation.

The primary advantage of our method is that the inpainting takes into account the information of all the remaining shape instead of only the vicinity of missing regions. Rather than directly estimating the missing geometry, we actually estimate the reconstruction coefficients of the whole original shape. Since the mesh Laplacian basis functions are smooth and have global support, the reconstructed inpainting shape is naturally smooth and globally coherent with a simple intrinsic structure.

The main contributions of this work are:

- We introduce a new surface inpainting framework based on representations in the transformed domain and sparsity constraints on reconstruction coefficients. This framework can make use of the information of the whole remaining shape and inpainting results tend to be simple and globally coherent.
- We study the sparsity of 3D shapes with respect to their Laplacian eigenbases and show their effectiveness in surface inpainting.
- We demonstrate the high performance of our inpainting method with several examples in hole filling and mesh editing.

2. Related work

2.1. Surface inpainting

Many methods have been proposed in the research literature dealing with the general problem of surface inpainting, bearing different names such as hole filling, mesh completion, and surface restoration. We refer readers to Attene et al. (2013) for a recent survey of popular algorithms for hole filling and mesh completion.

One simple approach for surface inpainting is by filling the missing region with an inpainting patch that interpolates the surrounding geometry. The interpolating patch may be generated with simple polynomial functions (Wang and Oliveira, 2007), triangular B-splines (Pfeifle and Seidel, 1996), or radial basis functions (Branch et al., 2006), and are generally smooth and continuous across the boundaries. The interpolation-based approaches, however, only work well with disk-like holes and are not suitable for filling regions with complex boundaries.

Typical mesh-based hole filling algorithms have two steps: (1) Find an initial triangulation of the missing region defined by the hole boundary; (2) Optimize the inserted mesh to improve its fairness and coherence with surrounding shapes. Liepa (2003) performed hole triangulation with a dynamic programming technique taking into account the dihedral angles and areas of the created triangles. The inserted mesh is then optimized with Laplacian smoothing to improve fairness. In Zhao et al. (2007), surface holes are patched by an advancing-front mesh generation method, and the vertex positions are optimized by solving a Poisson equation based on the desirable triangle normals computed from boundary vertices. In Bac et al. (2008), the coordinates of the inserted vertices are optimized by minimizing the discrete thin-plate energy. In Li et al. (2010), Wang et al. (2012) and Ngo and Lee (2013), complex holes are first partitioned into sub-holes by feature curves extended from the existing parts; typical hole-filling then can be performed on these sub-holes which are much more planar.

Another class of inpainting algorithms are based on variational methods. The basic idea is to iteratively evolve the inpainting shape by optimizing a functional that constrains certain geometric properties of the inserted mesh, e.g., positions, areas, tangency, and curvatures. Pernot et al. (2006) developed a hole filling algorithm which minimizes the variational involving curvature between the surrounding and inpainted geometry. In Caselles et al. (2008), the completing surface is chosen such that a power of the mean curvature is minimized. Clarenz et al. (2004) proposed a shape restoration algorithm by computing the l_2-gradient flow of the Willmore energy which ensures the continuity of the normal field.

Finally, a lot of mesh inpainting methods are exemplar-based. Kraevoy and Sheffer (2005) introduced an algorithm for mesh completion by mapping the incomplete mesh with a template model. Context-based surface completion methods like

Sharf et al. (2004) and Park et al. (2005) fill holes by importing patches from similar regions and adapt them to the hole regions via shape deformation. A coherence objective can be imposed such that every local neighborhood of the filled mesh is similar to some local neighborhood from the input mesh (Harary et al., 2014).

2.2. Sparse signal recovery and inpainting

Recent years have witnessed a surge in the research of sparsity-based signal recovery. The fundamental idea is that a sufficiently sparse signal can be reliably reconstructed from partial measurements by exploiting the sparsity cue.

Sparse signal recovery has seen most success in compressed sensing applications, where the measurement/sensing matrix is typically chosen as a normalized random matrix which satisfies the restricted isometry property with high probability. For the inpainting problem, the measurement is expressed as a mask matrix, which is not strictly a valid compressed sensing process. Nonetheless, we can still take advantage of the sparsity constraints to recover the original signals in many situations.

For image inpainting and restoration tasks, many algorithms based on sparse representation have been published. Guleryuz (2006a, 2006b) proposed an algorithm for image recovery based on adaptive sparse representation. In Elad et al. (2005), images are decomposed into texture and cartoon components, each of which is sparse with respect to a particular dictionary; the missing parts then can be easily reconstructed. Fadili and Starck (2005) and Fadili et al. (2009) formulated image inpainting as a maximum-likelihood estimation problem with a sparsity-promoting prior penalty imposed on the reconstructed coefficients. A similar formulation is proposed in Cai et al. (2008) where images have sparse framelet representations and the incomplete image can be restored via an iterative shrinkage algorithm. This formulation balances the sparsity of coefficients, fidelity to the existing data, as well as the smoothness of the solution. Ogawa and Haseyama (2011) proposed an image recovery algorithm based on sparse representation, in which the low-dimensional subspaces optimal for targeted missing textures are adaptively selected.

There have been very few works on the sparsity-induced recovery of signals defined on graphs. Zhu and Rabbat (2012) proposed to use the dictionary of graph Laplacian eigenfunctions to recover smooth and sparse graph signals and applied them to the reconstruction of wireless sensor networks data from partial node readings. To the best of our knowledge, our method is the first of such attempts to tackle the problem of geometry inpainting/completion via sparse signal recovery.

3. Variational inpainting model

The problem of mesh signal inpainting can be stated as follows. Consider a triangle mesh $M = \{V, E\}$ with n vertices, where V and E denote the set of vertices and edges, respectively. Let $\mathbf{f} \in \mathbb{R}^n$ be a vector signal defined on the mesh vertices. Assume the signal values at a subset of vertices $V' \subset V$ are already known, the goal of inpainting is to compute a reasonable estimate of the remaining signal values at $V - V'$. For the problem of inpainting surface geometry, the mesh signal is the coordinate function and the unknown parts correspond to surface holes.

Assume the number of known vertices is $|V'| = n'$. We can define the $n' \times n$ projection matrix P as

$$P(i, j) = \begin{cases} 1 & \text{if } v_j \text{ is the } i\text{th element of } V' \\ 0 & \text{otherwise.} \end{cases} \tag{1}$$

Denote the observable parts of $\mathbf{f}$ to be $\mathbf{f}' \in \mathbb{R}^{n'}$, which should satisfy $\mathbf{f}' = P\mathbf{f}$, the general inpainting problem can be formulated as a constrained optimization problem

$$\hat{\mathbf{f}} = \arg\min_{\mathbf{f}} \Pr(\mathbf{f}) \quad \text{s.t.} \quad \|P\mathbf{f} - \mathbf{f}'\|_2^2 < \epsilon, \tag{2}$$

or equivalently as a penalized maximum-likelihood estimation problem

$$\hat{\mathbf{f}} = \arg\min_{\mathbf{f}} \Pr(\mathbf{f}) + \lambda \|P\mathbf{f} - \mathbf{f}'\|_2^2, \tag{3}$$

where the data term $\|P\mathbf{f} - \mathbf{f}'\|_2^2$ emphasizes fidelity to the available observations, while $\Pr(\hat{f})$ is a prior regularizing certain properties of the reconstructed signal.

Traditionally, priors are chosen to optimize the fairness of the inserted mesh or its coherence with the surrounding geometry. For example, a commonly-adopted prior for surface optimization is $\Pr(\mathbf{f}) = \|L\mathbf{f}\|_2^2$ which aims to maximize the smoothness of the estimated signal, generating the so-called *least-squares meshes* (Sorkine and Cohen-Or, 2004). Here L is the Laplace operator of the shape.

Instead of computing the approximate signal $\hat{\mathbf{f}}$ in the mesh domain directly, we may first estimate the original signal's representation in some transformed domains. Consider a dictionary D of m atoms, where each atom is an elementary signal defined on the mesh; written in the matrix form, $D = (\mathbf{d_1}, \ldots, \mathbf{d_m})$, $\mathbf{d_i} \in \mathbb{R}^{n \times 1}$. The original signal f may be represented as the linear combination of columns in D

$$\mathbf{f} = D\alpha = \sum_{i=1}^{m} \alpha_i \mathbf{d_i}, \tag{4}$$

where $\alpha = (\alpha_1, \ldots, \alpha_m)^T$ is the coefficient representation of $\mathbf{f}$ w.r.t. the dictionary D.

Obviously, if we can estimate the coefficient representation of the whole original signal from partial measurements $\mathbf{f}'$, then we also obtain an inpainting of the missing signal values. If we know in advance that the coefficients of representation of $\mathbf{f}$ satisfy certain statistical properties, we can estimate the coefficients by imposing a prior on α

$$\hat{\alpha} = \arg \min_{\alpha} \Pr(\alpha) \quad \text{s.t.} \quad \|PD\alpha - \mathbf{f}'\|_2^2 < \epsilon. \tag{5}$$

The complete original signal can then be estimated as $\hat{\mathbf{f}} = D\hat{\alpha}$.

4. Sparsity-based surface inpainting

The fundamental idea of our sparsity-based surface inpainting method is that, for most natural shapes, although the surface geometry is a high-dimensional signal, it actually lives in a low-dimensional subspace and has a sparse representation in some transformed domains. Hence, we can set the sparsity of coefficients as the prior in Eq. (5) to estimate the coefficient representation of the global shape and recover the missing geometry. As long as the "complexity" of the original shape is much smaller than the number of available observations, we have a good chance to obtain a plausible restoration.

In this section, we first discuss the sparsity of shape geometry w.r.t. the mesh Laplacian eigenbasis, demonstrating the potentials of Laplacian eigenfunctions for sparsity-based geometry processing. Then we propose a sparsity-constrained formulation for the problem of surface inpainting with known connectivity. Finally, we extend our inpainting method to hole filling-in where mesh connectivity is nonexistent in the missing regions in the first place.

4.1. Laplacian eigenbasis

For a discrete mesh, its graph Laplacian matrix L is typically defined as

$$L(i, j) = \begin{cases} 1 & \text{if } (v_i, v_j) \in E \\ 0 & \text{otherwise.} \end{cases} \tag{6}$$

The set of eigenfunctions of L, $\Phi = \{\phi_i\}_{i=1}^n$, are commonly referred to as Laplacian eigenbasis or manifold harmonic basis (MHB) (Vallet and Lévy, 2008). The Laplacian eigenfunctions are analogous to the classic Fourier basis in Euclidean space and have the following similar properties:

- Functions in $\{\phi_i\}$ all have global support on the mesh.
- Functions in $\{\phi_i\}$ exhibit wave-like periodical oscillations on the mesh with different frequencies corresponding to the eigenvalues $\{\lambda_i\}$.
- $\{\phi_i\}$ form a complete, orthonormal basis of the square-integrable function space $L^2(M)$ defined on the mesh.
- $\{\phi_i\}$ induce a spectral transform: Any signal $f \in L^2(M)$ have a unique decomposition w.r.t. $\{\phi_i\}$

$$f = \sum_{k=1}^n \tilde{f}(k)\phi_k = \sum_{k=1}^n \langle f, \phi_k \rangle \phi_k,$$

in which $\tilde{f}(k)$ denotes the kth spectral/Fourier coefficient.

The aforementioned attractive properties make Laplacian eigenfunctions potentially efficient for representing shape signals defined on meshes. Karni and Gotsman (2000) utilized the truncated spectral coefficients for compressed representation of mesh geometry, which is very similar to the JPEG format for image compression. Ben-Chen and Gotsman (2005) further proved that the Laplacian eigenbasis is the optimal basis for mesh compression in the mean square error (MSE) sense, provided that the distribution of the vertex coordinates satisfy certain natural assumptions.

The spectral mesh compression method introduced in Karni and Gotsman (2000) basically computes the linear approximation of mesh geometry expanded on its Laplacian eigenbasis. In linear approximation, spectral coefficients are always added from low-frequency to high-frequency, regardless of their respective contributions to the original signal. Better coefficient sparsity can be achieved through nonlinear approximation by prioritizing coefficients of larger magnitude.

As an example, Fig. 1 shows the power of Laplacian eigenbasis for shape approximation. Figs. 1(b)–1(c) visualize the mesh coordinate functions of the example mesh and their spectral transform coefficients, respectively. We can easily see that the coordinate functions have very dense support in the natural graph basis, but can be sparsely represented in the spectral/Fourier domain in the sense that the majority of spectral coefficients are almost 0. Moreover, the few significant coefficients are mostly concentrated in the low-frequency end, especially for a smooth shape model.

Fig. 1(d) shows the spectral energies contained in the first k coefficients (linear approximation) and the first k most significant coefficients (nonlinear approximation). We see that the vast majority of spectral energy are captured by the first few significant coefficients.

Fig. 1(e) shows how the mesh reconstruction error changes with the number of coefficients being used. In this example, we see that the approximation error becomes negligible using only about 20 non-zero coefficients.
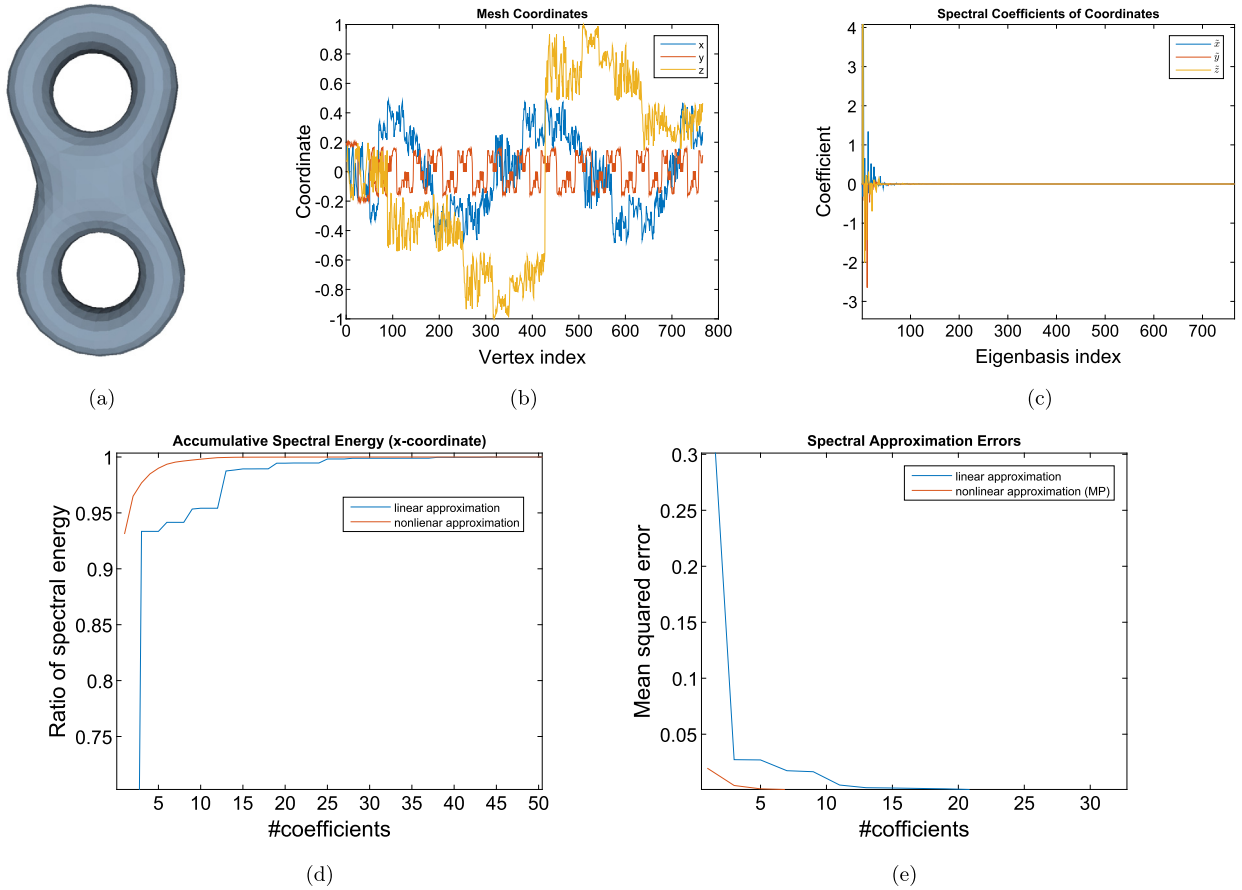
Fig. 1. Approximation of the double-torus model with the Laplacian eigenbasis. (a) Original shape; (b) Vertex coordinate functions; (c) Spectral coefficients of the coordinate functions w.r.t. the Laplacian eigenbasis; (d) Ratios of spectral energy contained in the first k coefficients; (e) Approximation error of the mesh geometry using the first k coefficients.

4.2. Surface inpainting

In the previous section, we have shown that the geometry of a 3D shape generally has a sparse representation w.r.t. its Laplacian eigenbasis. Hence, we can set the Laplacian eigenvector as the reconstruction dictionary and use the sparsity of coefficients as a prior to estimate the representation of missing shape geometry.

Following the formulation in Sec. 3, surface inpainting can be rewritten as the following sparse approximation problem

$$\hat{\alpha} = \arg\min_{\alpha} \|\alpha\|_0 \quad \text{s.t.} \quad \|P\Phi\alpha - \mathbf{x}'\|_2^2 < \epsilon, \tag{7}$$

$$\hat{\mathbf{x}} = \Phi\alpha. \tag{8}$$

Here the pseudo-norm $\|\alpha\|_0 = \#\{i : \alpha_i \neq 0\}$ denotes the support of α, which counts the number of non-zero components of α, Φ denotes the dictionary matrix comprising the Laplacian eigenfunctions, and $\hat{\mathbf{x}}$ and $\mathbf{x}'$ represent the estimated and observable coordinate functions, respectively.

We should note that, since the Laplacian eigenbasis constitute a complete dictionary, Eq. (7) is solvable even if we set $\epsilon = 0$, in which case the reconstructed shape will exactly match the known geometry. However, strictly sparse signals are rare in real life. It is much more likely that the unknown shape geometry is *compressible* or *weakly sparse* w.r.t. the dictionary of Laplacian eigenvectors, i.e., the nonlinear approximation errors observe a power law decay as the number of participating basis vectors increases (Starck et al., 2010). In practice, we set $\epsilon > 0$ to trade off exact reproduction for a sparser α, i.e., allowing the reconstructed signal to have small discrepancies with the observation.

Solving l_0 optimization is an NP-hard problem in nature. Fortunately, under certain conditions, greedy algorithms such as orthogonal matching pursuit (OMP) and its variants can generate the exact sparse solution or a good enough approximation (Needell and Vershynin, 2010; Tropp and Gilbert, 2007).
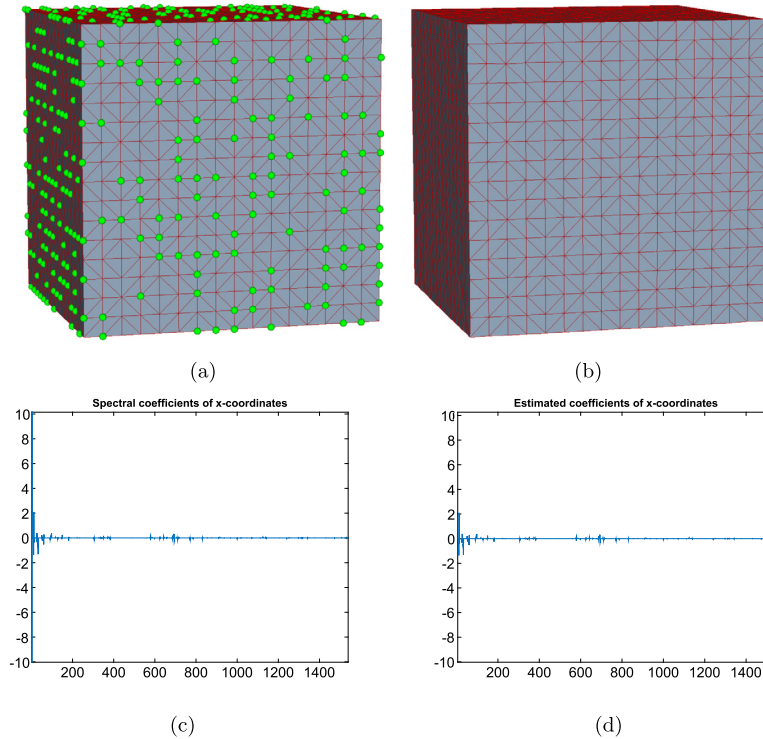
Fig. 2. Estimating the Laplacian eigenbasis coefficients of the cube model with 40% random missing vertices. (a) Original shape model; green dots denote vertices that are labeled as missing. (b) The reconstructed shape using our inpainting method. (c) The coefficient representation of the original shape's x-coordinates. (d) Estimated coefficient representation of the x-coordinates inferred from the information of available vertices. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Another approach to find an approximated solution to Eq. (7) is to relax the highly discontinuous l_0 norm with l_1 norm, i.e.

$$\hat{\alpha} = \arg \min_{\alpha} \|\alpha\|_1 \quad \text{s.t.} \quad \|P\Phi\alpha - \mathbf{x}'\|_2^2 < \epsilon, \tag{9}$$

or equivalently,

$$\hat{\alpha} = \arg \min_{\alpha} \|P\Phi\alpha - \mathbf{x}'\|_2^2 + \lambda \|\alpha\|_1. \tag{10}$$

The estimation problem then becomes convex and solvable. There are several readily available algorithms for solving l_1 optimization, e.g., interior point method (Kim et al., 2007), iteratively reweighted least squares (IRLS) (Holland and Welsch, 1977), least angle regression (LARS) (Sorkine and Cohen-Or, 2004), and iterative shrinkage-thresholding (Daubechies et al., 2004).

For the task of surface inpainting, we find that l_1 optimization algorithms tend to be more robust and generally produce better inpainting results than greedy algorithms. In this work, we use the l1_ls solver introduced in Kim et al. (2007) which implements a fast interior-point method for solving l_1-regularized least-square problems like Eq. (10).

As an example, Fig. 2 demonstrates the potentials of our sparsity-based inpainting method. We randomly label 40% of vertices of the original cube model as missing vertices, and use the coordinates of the remaining vertices to estimate the spectral coefficient representation of the original shape by solving Eq. (9). Fig. 2(b) shows the shape reconstructed from the estimated spectral coefficients. Figs. 2(c)–2(d) show the spectral coefficients computed from the original x-coordinate function and the coefficients estimated by our sparsity-based method, respectively. In this example, our method recovers the sparse coefficient representation in a very precise way.

4.3. Filling surface holes

One of the most important technical elements of our surface inpainting method is the dictionary of global shape basis, which are determined by the global mesh connectivity. For some applications such as repairing damaged surface regions, the mesh connectivity of the region to be repaired is already known in advance before reconstruction and we may not need to modify it. For hole filling applications, however, the inpainting regions are completely blank without any inside

information. It is imperative to establish interior mesh connectivity, by way of vertex insertion and patch triangulation, before our surface inpainting method can be applied.

Obviously, how a patch (to be used to cover the hole region) is triangulated directly influences the final inpainting result in our framework. In general, a good patch triangulation should ensure the vertex density of the inserted mesh to be consistent with the remaining mesh. In this paper, we adopt the algorithms proposed by Liepa (2003) for hole triangulation and refinement. Algorithm 1 summarizes the pipeline of our sparsity-based hole filling method.

Algorithm 1 Sparsity-based hole filling.

Require: Input mesh M
1: Identify surface holes,
2: Triangulate and refine holes using the algorithms described in Liepa (2003),
3: Compute the mesh Laplace matrix L and the Laplacian eigenbasis dictionary Φ,
4: **for** coordinate **x**, **y**, and **z do**
5:    Compute the spectral representation α of the global coordinates by solving Eq. (9),
6:    Reconstruct the coordinates of the inserted mesh with $\Phi\alpha$,
7: **end for**

4.4. Remarks on dictionary

Although the dictionary of Laplacian eigenvectors in general has strong compressive power for encoding shape geometry, it also has some limitations. Similar to Fourier basis, the Laplacian eigenvectors are most suitable for representing smooth signals or globally repetitive features, but are generally not optimized for encoding shapes with many local sharp features. In the image domain, other than 2D Fourier basis, people have developed various types of harmonic basis (e.g., wavelet, curvelet, ridgelet, etc.) for efficient encoding of images of different properties. For example, the ridgelets are especially efficient in representing piecewise smooth images with global straight edges (Fadili and Starck, 2012). In the mesh domain, however, we do not have such diverse harmonic basis to choose from, which for now limits the power of sparsity-based methods.

Another issue is related to the ratio of Laplacian eigen-decomposition. Computing the full set of Laplacian eigenvectors of a large mesh is extremely time consuming, generally infeasible for meshes with more than a few thousand vertices on a regular PC. Fortunately, for our surface inpainting applications, it is actually not necessary or even desirable to compute the full set of eigenvectors. On the one hand, for smooth shapes, the spectral energy is overwhelmingly concentrated on the low-frequency end, and a dictionary composed of only low-frequency eigenvectors can well approximate the shape geometry with very little error. On the other hand, the high-frequency Laplacian eigenvectors are less stable than the low-frequency ones, and including them in the dictionary may cause overfitting and result in worse inpainting results, since the high-frequency eigenvectors are more correlated with local geometric details than with the overall structure of the shape. In our experiments, we find that the best inpainting results are usually achieved with a dictionary of 20% to 50% total eigenvectors.

5. Experiments

In this section, we first evaluate the performance of our sparsity-based inpainting algorithms on recovering missing geometry from partial observations. Then we demonstrate how our method can be applied to repairing damaged geometry and filling surface holes.
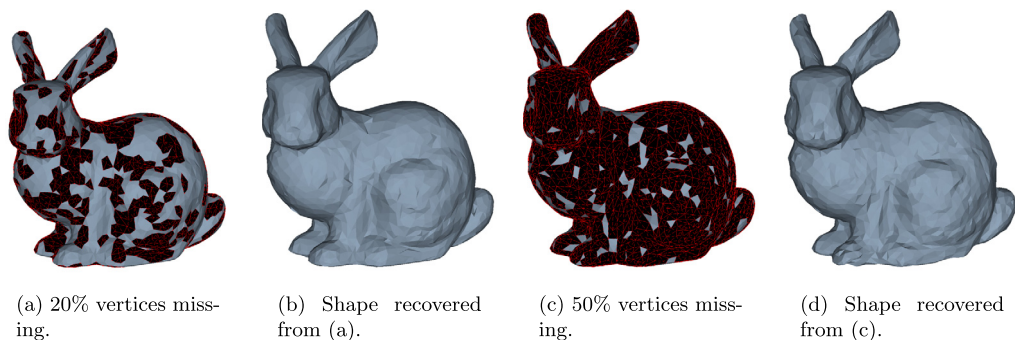

(a) 20% vertices missing. (b) Shape recovered from (a). (c) 50% vertices missing. (d) Shape recovered from (c).

Fig. 3. Recovery of the bunny model with 20% and 50% random missing vertices.

(a) 20% vertices missing.

(b) Shape recovered from (a).

(c) 50% vertices missing.

(d) Shape recovered from (c).
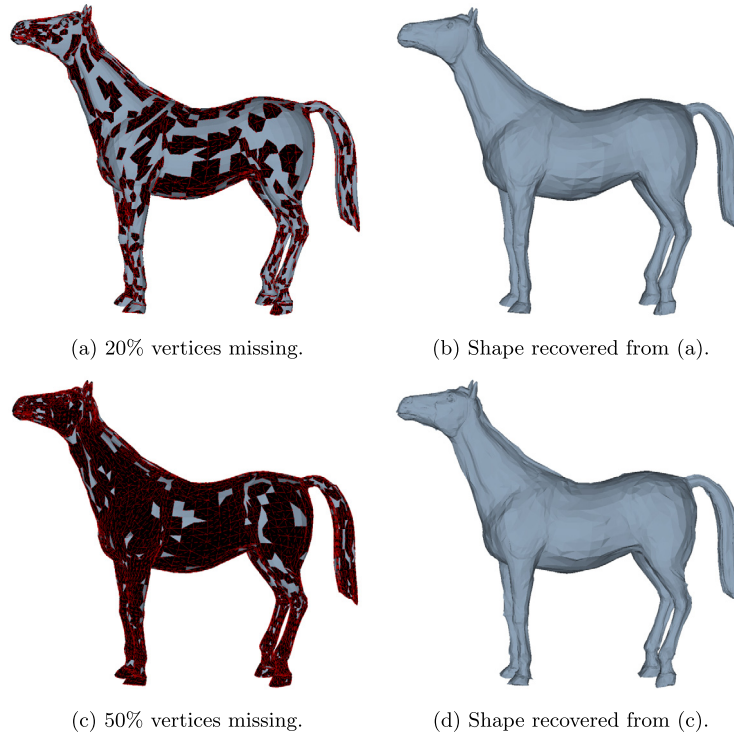
Fig. 4. Recovery of the horse model with 20% and 50% random missing vertices.

(a) 20% vertices missing.

(b) Shape recovered from (a).

(c) 50% vertices missing.

(d) Shape recovered from (c).
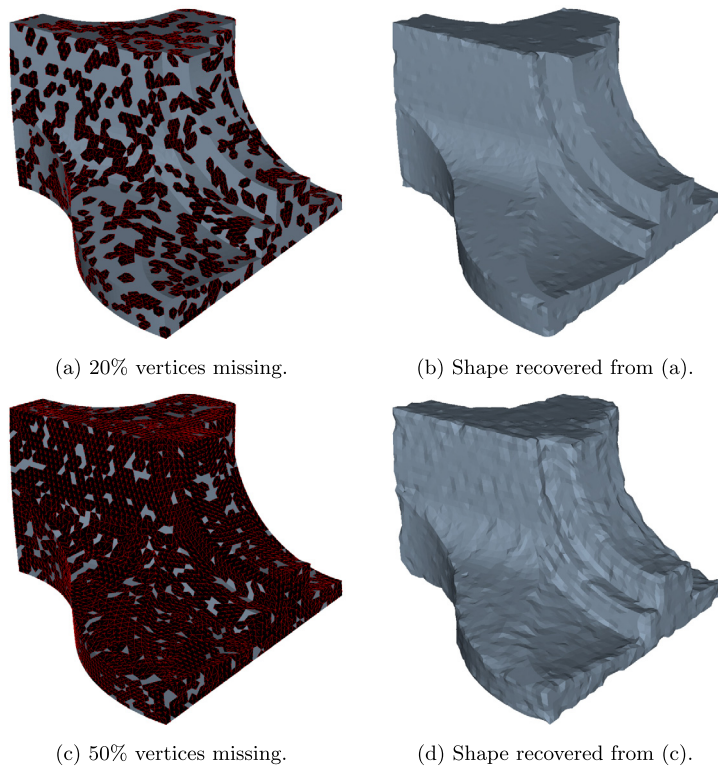
Fig. 5. Recovery of the fandisk model with 20% and 50% random missing vertices.

(a) 20% vertices missing.

(b) Shape recovered from (a).

(c) 50% vertices missing.

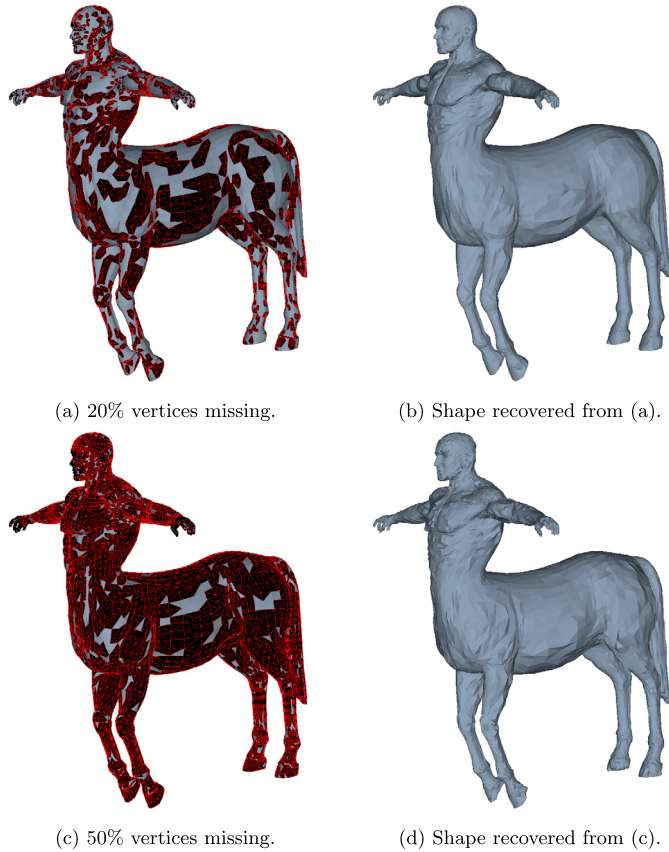(d) Shape recovered from (c).

Fig. 6. Recovery of the centaur model with 20% and 50% random missing vertices.
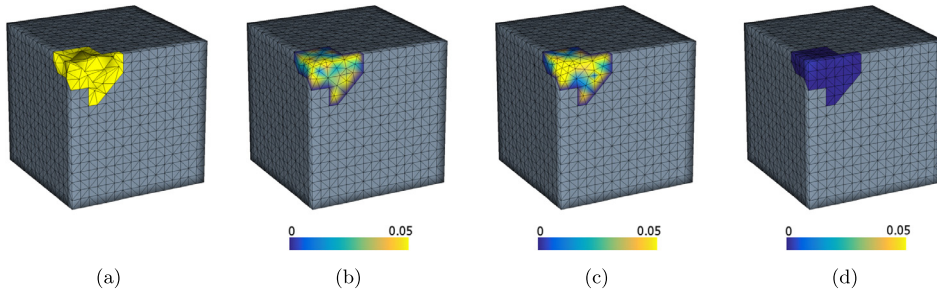

(a) (b) (c) (d)

Fig. 7. Geometry repair of the cube model by replacing the selected damaged regions (marked in yellow) with an inpainting patch. (a) The damaged model; (b) Repaired with Laplacian regularized least square smoothing (Nealen et al., 2006); (c) Repaired with thin-plate energy minimization (Bac et al., 2008); (d) Repaired with our inpainting method. In (b)–(d), the per-vertex error (compared with the ground truth) is color-coded. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 1

Geometry recovery errors and time performance. For each model, we test the recovery performance with 20% and 50% randomly selected vertices labeled as missing. Each experiment has been repeated three times and averaged on a system with quad-core 2.4 GHz CPU and 16 GB RAM.

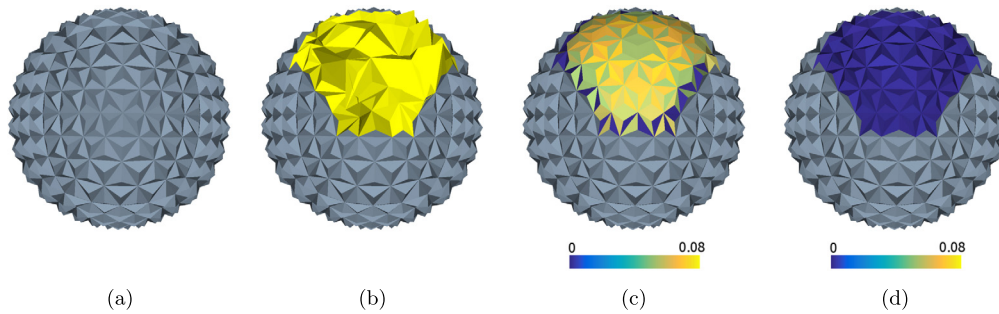| Mesh	#vertices	#eigenvectors	Decomposition time (s)	Missing ratio	Error	l_1 time (s)
bunny	2.5k	1000	19.6	0.2	1.9e−2	2.8
				0.5	2.2e−2	2.1
horse	8.4k	4000	1015.4	0.2	8.6e−3	19.7
				0.5	1.0e−2	59.0
fandisk	6.5k	3000	445.2	0.2	8.8e−3	17.4
				0.5	1.2e−2	33.9
centaur	15.8k	2000	397.7	0.2	7.1e−3	12.2
				0.5	8.0e−3	15.0

Fig. 8. Geometry repair of the damaged epcot model. (a) The original epcot model; (b) The damaged model (damaged region is marked in yellow); (c) Repaired with thin-plate energy minimization (Bac et al., 2008); (d) Repaired with our inpainting method. In (c) and (d), the per-vertex error (compared with the ground truth) is color-coded. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)
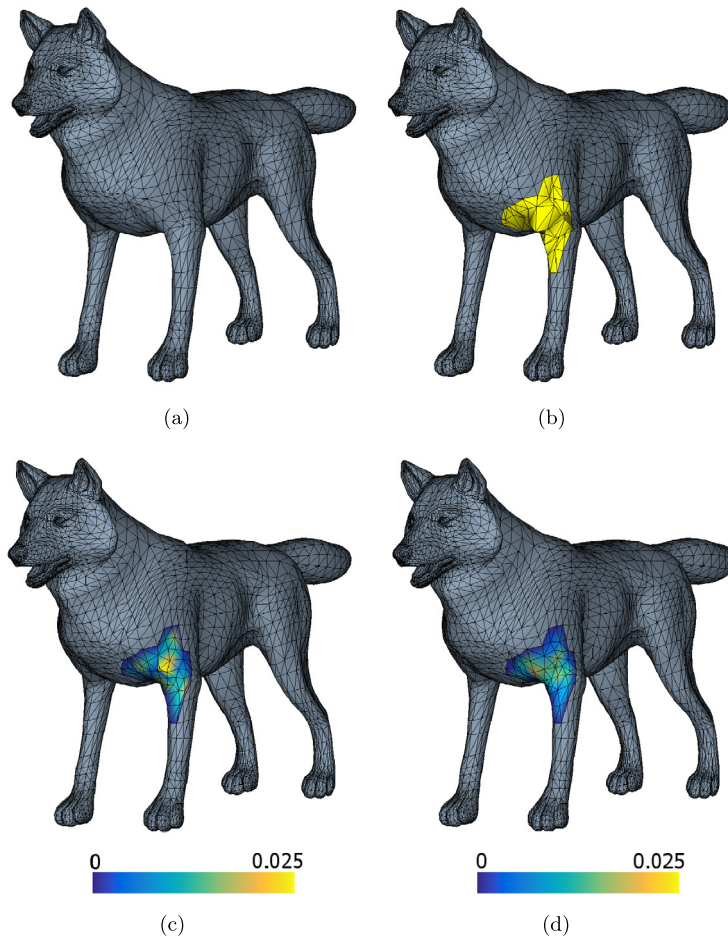

Fig. 9. Geometry repair of the damaged wolf model. (a) The original wolf model; (b) Damaged model with significant noise in the region marked in yellow; (c) Repaired with Laplacian regularized least square smoothing (Nealen et al., 2006); (d) Repaired with our inpainting method. In (c) and (d), the per-vertex inpainting error (compared with the ground truth) is color-coded. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

5.1. Geometry recovery

To evaluate the performance of geometry recovery, for each testing model, we randomly label 20%–50% vertices as missing and use our sparsity-based inpainting method to estimate the original geometry based on the coordinates of the still available vertices. The estimated coordinates are then compared with the original coordinates.
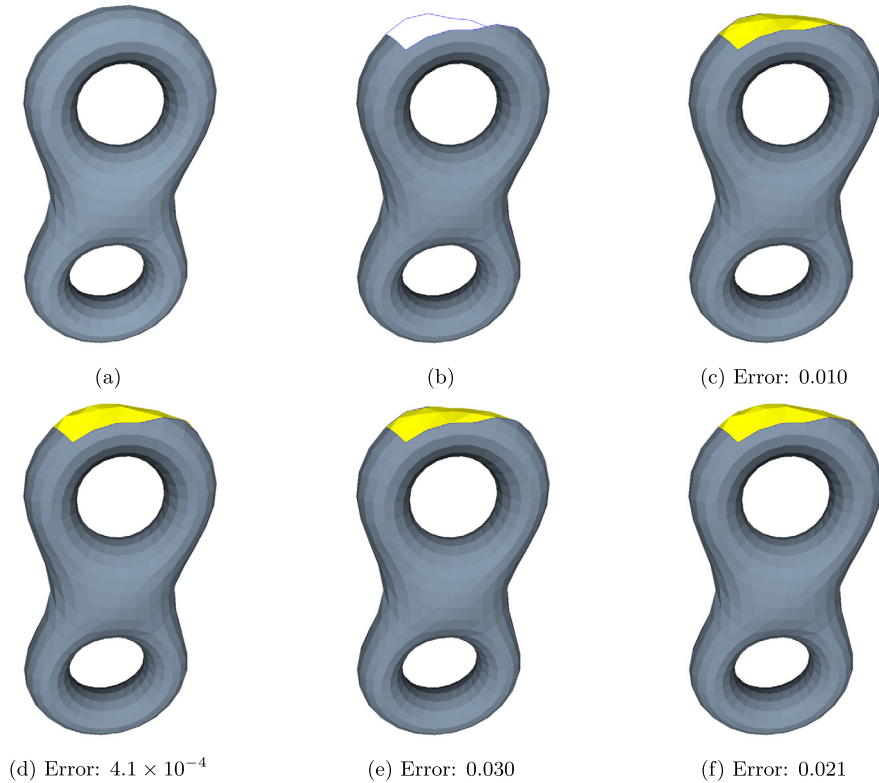
Fig. 10. Compare hole filling results using our sparsity-regularized method and the geometry-regularized method introduced in Bac et al. (2008). The error is measured as the root-mean-squared deviation from the estimated vertices to the original shape. (a) The original double torus model. (b) The model with a hole. (c) Inpainted using the method in Bac et al. (2008) with the original mesh connectivity. (d) Inpainted using our method with the original mesh connectivity. (e) Inpainted using the method in Bac et al. (2008) with the mesh connectivity generated from hole triangulation and refinement. (f) Inpainted using our method with the same mesh connectivity as (e).

All the testing models have been translated and scaled to be contained inside the unit cube. The recovery error is measured as the root-mean-square error (RMSE) of the coordinates of the missing vertices.

Fig. 3, Fig. 4, Fig. 5, and Fig. 6 show some examples of geometry recovery with 20% and 50% missing vertices. Table 1 documents the recovery errors and time performance of our tests. From the experimental results, we have the following observations:

- When the missing vertices are randomly dispersed on the shape, our sparsity-based method can reliably recover the missing coordinates with great precision, even when the ratios of missing vertices are as high as 50%.
- The l_1 estimation generally becomes more time consuming when the ratio of missing vertices increases.
- As noted in Sec. 4.4, using the truncated Laplacian eigenbasis dictionary is acceptable for restoring smooth shapes. However, for shapes with many edges and corners, such as the fandisk model (see Fig. 5), our inpainting method cannot well preserve local discontinuities, since the high-frequency basis are simply not present in the truncated dictionary.

5.2. Geometry repair

Our sparsity-based inpainting method is very suitable for repairing partially damaged geometry. After manually selecting the damaged regions, we can apply our inpainting method to estimate the original whole shape with the same connectivity based on the remaining parts of the shape. The corrupted regions can then be substituted by the inpainting patch.

Figs. 7, 8, and 9 demonstrate repairing damaged local geometry using our sparsity-regularized inpainting method. The results are compared with two geometry-regularized mesh optimization methods: Laplacian regularized least square smoothing (Nealen et al., 2006) and thin-plate energy minimization (Bac et al., 2008). We can see that, although geometry-regularized methods can generate patches that are smooth and blend well with the surroundings, they fail to recognize the intrinsic structures of the original shapes; consequently, important geometric features are simply smoothed out. In contrast, our sparsity-regularized inpainting method takes into account the global shape structures, and almost perfectly recovers the edges and corners in the cube model (Fig. 7) and the geometric textures of the epcot model (Fig. 8) from partial observations.
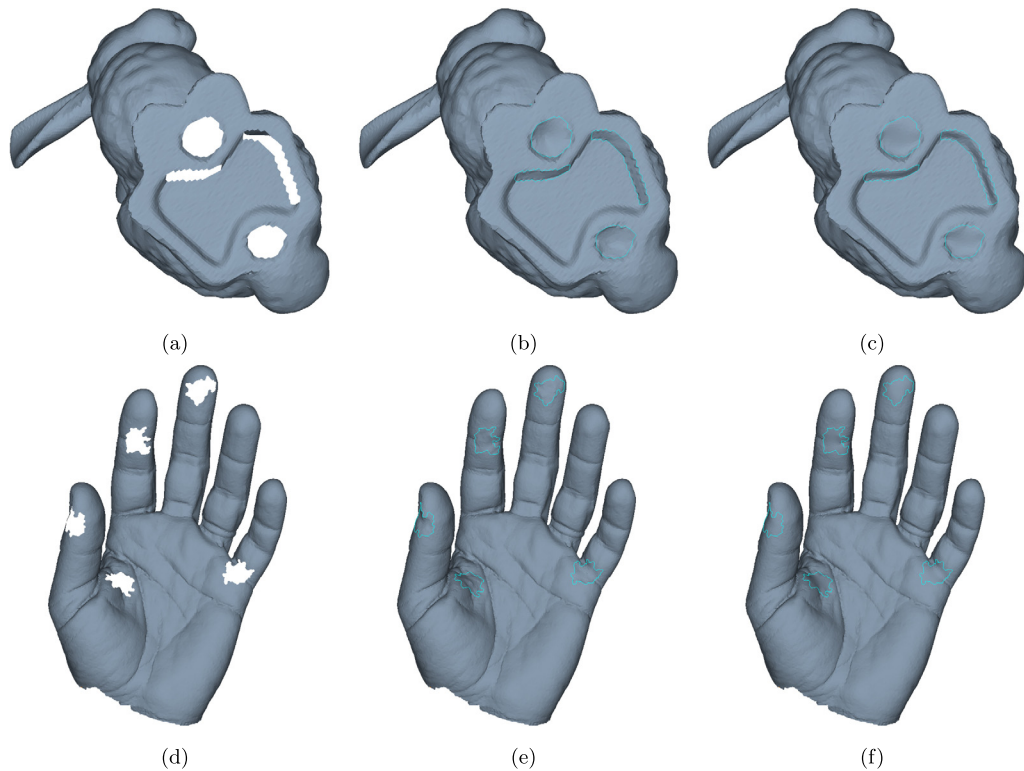
Fig. 11. Inpainting existing holes on the bunny and hand models. (a), (d) Original models with holes; (b), (e) Hole-filling result using our method; (c), (f) Hole-filling result using the method of Bac et al. (2008).

5.3. Hole filling

As introduced in Sec. 4.3, for general hole filling tasks, the mesh connectivity information inside holes are probably unknown. We must first triangulate holes in a proper way and then apply our geometry inpainting method to optimize the newly inserted mesh. How the hole is triangulated directly impacts the global Laplacian eigenbasis which subsequently determine the estimated recovery.

As an example, Fig. 10 compares the results of filling the holes of a double torus model with and without original connectivity information, using our sparsity-regularized method and the geometry-regularized method proposed in (Bac et al., 2008). We can see that estimating with a different connectivity significantly alters the final hole fairing results. In this example, our method generate shapes that are more approximate to the original shape both with the original connectivity and with the new connectivity.

In most cases, we cannot expect the hole filling result using our sparsity-regularized inpainting method to precisely match the original shape when the number of vertices and connectivity of the patching mesh, generated from hole triangulation and refinement, are different from the original mesh. Nonetheless, the resulting patching meshes tend to be coherent with the whole remaining shape, thanks to the global shape awareness of our method. Fig. 11 shows two examples of filling holes utilizing our inpainting method. The results are comparable to the geometry-regularized surface restoration method in Bac et al. (2008).

6. Conclusion

In this paper, we have proposed a novel surface inpainting algorithm based on sparse signal recovery. Instead of directly estimating the local missing geometry, our new inpainting framework is designed to discover the coefficient representation of the entire original shape in a transform domain. When the shape geometry is sufficiently sparse with respect to the dictionary of transform basis, chances are we can accurately recover this sparse representation by imposing sparsity constraints on the coefficients given partial observations. In our method, we adopt the mesh Laplacian eigenbasis as dictionary, and formulate surface inpainting as a sparse signal recovery problem. Leveraging standard l_1 optimization techniques, we can obtain an estimated shape which agrees with the observable parts and are globally coherent. For shapes that are highly compressible w.r.t. the Laplacian eigenbasis, we have experimentally demonstrated the great potential of our method for geometry restoration, geometry repair, and hole filling.

For the future work, we plan to extend our sparsity-based inpainting framework by integrating geometric constraints such as curvatures and normals, which should improve the geometric consistency of the inpainting result. We are also interested in designing new types of shape basis and exploring more sophisticated strategies for constructing dictionaries, e.g., dictionaries that are adaptive to the input shape.

Acknowledgements

References

Attene, M., Campen, M., Kobbelt, L., 2013. Polygon mesh repairing: an application perspective. ACM Comput. Surv. 45 (2), 15.

Bac, A., Tran, N.-V., Daniel, M., 2008. A multistep approach to restoration of locally undersampled meshes. In: Proceedings of the 5th International Conference on Advances in Geometric Modeling and Processing. GMP'08. Springer-Verlag, Berlin, Heidelberg, pp. 272–289.

Ben-Chen, M., Gotsman, C., 2005. On the optimality of spectral compression of mesh data. ACM Trans. Graph. 24 (1), 60–80.

Branch, J., Prieto, F., Boulanger, P., 2006. Automatic hole-filling of triangular meshes using local radial basis function. In: Third International Symposium on 3D Data Processing, Visualization, and Transmission. IEEE, pp. 727–734.

Cai, J.-F., Chan, R.H., Shen, Z., 2008. A framelet-based image inpainting algorithm. Appl. Comput. Harmon. Anal. 24 (2), 131–149.

Candes, E.J., Romberg, J.K., Tao, T., 2006. Stable signal recovery from incomplete and inaccurate measurements. Commun. Pure Appl. Math. 59 (8), 1207–1223.

Caselles, V., Haro, G., Sapiro, G., Verdera, J., 2008. On geometric variational models for inpainting surface holes. Comput. Vis. Image Underst. 111 (3), 351–373.

Clarenz, U., Diewald, U., Dziuk, G., Rumpf, M., Rusu, R., 2004. A finite element method for surface restoration with smooth boundary conditions. Comput. Aided Geom. Des. 21 (5), 427–445.

Daubechies, I., Defrise, M., De Mol, C., 2004. An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. Commun. Pure Appl. Math. 57 (11), 1413–1457.

Elad, M., Starck, J.-L., Querre, P., Donoho, D.L., 2005. Simultaneous cartoon and texture image inpainting using morphological component analysis (MCA). Appl. Comput. Harmon. Anal. 19 (3), 340–358.

Fadili, M.-J., Starck, J.-L., 2005. EM algorithm for sparse representation-based image inpainting. In: IEEE International Conference on Image Processing, vol. 2. ICIP 2005. IEEE. pp. II-61.

Fadili, J., Starck, J.-L., 2012. Curvelets and ridgelets. In: Computational Complexity. Springer, pp. 754–773.

Fadili, M.-J., Starck, J.-L., Murtagh, F., 2009. Inpainting and zooming using sparse representations. Comput. J. 52 (1), 64–79.

Guleryuz, O.G., 2006a. Nonlinear approximation based image recovery using adaptive sparse reconstructions and iterated denoising – Part I: theory. IEEE Trans. Image Process. 15 (3), 539–554.

Guleryuz, O.G., 2006b. Nonlinear approximation based image recovery using adaptive sparse reconstructions and iterated denoising – Part II: adaptive algorithms. IEEE Trans. Image Process. 15 (3), 555–571.

Harary, G., Tal, A., Grinspun, E., 2014. Context-based coherent surface completion. ACM Trans. Graph. 33 (1), 5.

Holland, P.W., Welsch, R.E., 1977. Robust regression using iteratively reweighted least-squares. Commun. Stat., Theory Methods 6 (9), 813–827.

Karni, Z., Gotsman, C., 2000. Spectral compression of mesh geometry. In: Proceedings of the 27th annual conference on Computer graphics and interactive techniques. ACM Press/Addison–Wesley Publishing Co., pp. 279–286.

Kim, S.-J., Koh, K., Lustig, M., Boyd, S., Gorinevsky, D., 2007. An interior-point method for large-scale ℓ_1-regularized least squares. IEEE J. Sel. Top. Signal Process. 1 (4), 606–617.

Kraevoy, V., Sheffer, A., 2005. Template-based mesh completion. In: Symposium on Geometry Processing, pp. 13–22.

Li, Z., Meek, D.S., Walton, D.J., 2010. Polynomial blending in a mesh hole-filling application. Comput. Aided Des. 42 (4), 340–349.

Liepa, P., 2003. Filling holes in meshes. In: Proceedings of the 2003 Eurographics/ACM SIGGRAPH Symposium on Geometry Processing. Eurographics Association, pp. 200–205.

Nealen, A., Igarashi, T., Sorkine, O., Alexa, M., 2006. Laplacian mesh optimization. In: Proceedings of the 4th international conference on Computer graphics and interactive techniques in Australasia and Southeast Asia. ACM, pp. 381–389.

Needell, D., Vershynin, R., 2010. Signal recovery from incomplete and inaccurate measurements via regularized orthogonal matching pursuit. IEEE J. Sel. Top. Signal Process. 4 (2), 310–316.

Ngo, H.T.-M., Lee, W.-S., 2013. Feature-first hole filling strategy for 3D meshes. In: Computer Vision, Imaging and Computer Graphics. Theory and Applications. Springer, pp. 53–68.

Ogawa, T., Haseyama, M., 2011. Missing image data reconstruction based on adaptive inverse projection via sparse representation. IEEE Trans. Multimed. 13 (5), 974–992.

Park, S., Guo, X., Shin, H., Qin, H., 2005. Shape and appearance repair for incomplete point surfaces. In: Tenth IEEE International Conference on Computer Vision, 2005, vol. 2. ICCV 2005. IEEE, pp. 1260–1267.

Pernot, J.-P., Moraru, G., Véron, P., 2006. Filling holes in meshes using a mechanical model to simulate the curvature variation minimization. Comput. Graph. 30 (6), 892–902.

Pfeifle, R., Seidel, H.-P., 1996. Triangular b-splines for blending and filling of polygonal holes. In: Proceedings of the Conference on Graphics Interface'96. Canadian Information Processing Society, pp. 186–193.

Sharf, A., Alexa, M., Cohen-Or, D., 2004. Context-based surface completion. ACM Trans. Graph. 23, 878–887.

Sorkine, O., Cohen-Or, D., 2004. Least-squares meshes. In: Shape Modeling Applications, 2004. Proceedings. IEEE, pp. 191–199.

Starck, J.-L., Murtagh, F., Fadili, J.M., 2010. Sparse Image and Signal Processing: Wavelets, Curvelets, Morphological Diversity. Cambridge University Press.

Tropp, J.A., Gilbert, A.C., 2007. Signal recovery from random measurements via orthogonal matching pursuit. IEEE Trans. Inf. Theory 53 (12), 4655–4666.

Vallet, B., Lévy, B., 2008. Spectral geometry processing with manifold harmonics. Comput. Graph. Forum 27, 251–260.

Wang, J., Oliveira, M.M., 2007. Filling holes on locally smooth surfaces reconstructed from point clouds. Image Vis. Comput. 25 (1), 103–113.

Wang, X., Liu, X., Lu, L., Li, B., Cao, J., Yin, B., Shi, X., 2012. Automatic hole-filling of CAD models with feature-preserving. Comput. Graph. 36 (2), 101–110.

Zhao, W., Gao, S., Lin, H., 2007. A robust hole-filling algorithm for triangular mesh. Vis. Comput. 23 (12), 987–997.

Zhu, X., Rabbat, M., 2012. Graph spectral compressed sensing for sensor networks. In: IEEE International Conference on Acoustics, Speech and Signal Processing. ICASSP 2012. IEEE, pp. 2865–2868.