

# Pre-image as Karcher Mean Using Diffusion Maps: Application to Shape and Image Denoising

Nicolas Thorstensen, Florent Segonne, and Renaud Keriven

Universite Paris-Est, Ecole des Ponts ParisTech, Certis  
thorstensen@certis.enpc.fr  
<http://certis.enpc.fr/~thorsten>

**Abstract.** In the context of shape and image modeling by manifold learning, we focus on the problem of denoising. A set of shapes or images being known through given samples, we capture its structure thanks to the Diffusion Maps method. Denoising a new element classically boils down to the key-problem of *pre-image determination*, i.e. recovering a point, given its embedding. We propose to model the underlying manifold as the set of Karcher means of close sample points. This non-linear interpolation is particularly well-adapted to the case of shapes and images. We define the pre-image as such an interpolation having the targeted embedding. Results on synthetic 2D shapes and on real 2D images and 3D shapes are presented and demonstrate the superiority of our pre-image method compared to several state-of-the-art techniques in shape and image denoising based on statistical learning techniques.

## 1 Introduction

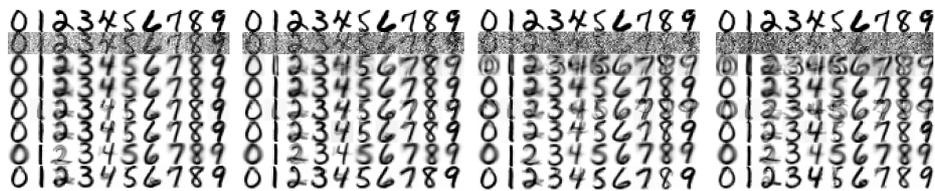
Manifold learning, the process of extracting the meaningful structure and correct geometric description present in a set of training points  $\Gamma = \{s_1 \cdots s_p\} \subset \mathbb{S}$ , has seen renewed interest over the past years. These techniques are closely related to the notion of dimensionality reduction, i.e. the process of recovering the underlying low dimensional structure of a manifold  $\mathcal{M}$  that is embedded in a higher-dimensional space  $\mathbb{S}$ . Among the most recent and popular techniques are the Locally Linear Embedding (LLE) [5], Isomap [6], Laplacian eigenmaps [7] and Diffusion Maps [8, 9, 10]. In this paper we focus on Diffusion Maps. Their nonlinearity, as well as their locality-preserving property and stable behavior under noise are generally viewed as a major advantage over classical methods like principal component analysis (PCA) and classical multidimensional scaling [8]. This method considers an adjacency graph on the set  $\Gamma$  of training samples, which matrix  $(W_{i,j})_{i,j \in 1, \dots, p}$  captures the local geometry of  $\Gamma$  - its local connectivity - through the use of a kernel function  $w$ .  $W_{i,j} = w(s_i, s_j)$  measures the strength of the edge between  $s_i$  and  $s_j$ . Typically  $w(s_i, s_j)$  is a decreasing function of the distance  $d_{\mathbb{S}}(s_i, s_j)$  between the training points  $s_i$  and  $s_j$ . In this work, we use the Gaussian kernel  $w(s_i, s_j) = \exp(-d_{\mathbb{S}}^2(s_i, s_j)/2\sigma^2)$ , with  $\sigma$  estimated as the median of the distances between all the training points [2, 10].

The kernel function has the property to implicitly map data points into a high-dimensional space, called the feature space. This space is better suited for the study of non-linear data. Computing the Diffusion Maps amounts to embed the data into the

feature space through a mapping  $\Psi$ . While the mapping from input space to feature space is of primary importance, the reverse mapping from feature space back to input space (the pre-image problem) is also useful. Consider for example the use of kernel PCA for pattern denoising. Given some noisy patterns, kernel PCA first applies linear PCA on the  $\Psi$ -mapped patterns in the feature space, and then performs denoising by projecting them onto the subspace defined by the leading eigenvectors. These projections, however, are still in the feature space and have to be mapped back to the input space in order to recover the denoised patterns.

**1.1 Related Work**

Statistical methods for shape processing are very common in computer vision. A seminal work in this direction was published by Leventon et. al. [11] adding statistical knowledge into energy based segmentation methods. Their method captures the main modes of variation by performing a PCA on the set of shapes. This was extended to non-linear statistics by Cremers et al. in [12]. The authors introduce non linear shape priors by using a probabilistic version of Kernel PCA (KPCA). Dambreville et.al [1] and Arias et al. [2] developed a method for shape denoising based on Kernel PCA. So did Kwok et al. [3] in the context of image denoising. Both methods compute a projection of the noisy datum onto a low dimensional space. In [13,4] the authors propose another kernel method for data denoising, the so called *Laplacian Eigenmaps Latent Variable Model* (LELVM), a probabilistic method. This model provides a dimensionality reduction and reconstruction mapping based on linear combinations of input samples. LELVM performs well on motion capture data but fails on complex shapes (see Fig. 1). Further we would like to mention the work of Pennec [14] and Fletcher [15] modeling the manifold of shapes as a Riemannian manifold and the mean of such shapes as a Karcher mean [16]. Their methodology is used in the context of computational anatomy to solve the average template matching problem. Closer to our work is the algorithm proposed by Etyngier et. al. [17]. They use Diffusion Maps as a statistical framework for non linear shape priors in segmentation. They augment an energy functional by a shape prior term. Contrary to us, they do not compute a denoised shape but propose an additional force toward a rough estimate of it.



**Fig.1.** Digit images corrupted by additive Gaussian noise (from left to right,  $\sigma^2 = 0.25, 0.45, 0.65, 0.85$ ). The different rows respectively represent, from top to bottom: the original digits; the corrupted digits; denoising with [1]; with [1]+ [2]; with [3]; with [3]+ [2]; with [4]; with our Karcher means based method. See table 2 for quantified results.

## 1.2 Our Contributions

In this paper, we propose a new method to solve the pre-image (see Section 3) problem in the context of Diffusion Maps for shape and image denoising. We suggest a manifold interpretation and learn the intrinsic structure of a given training set. Our method relies on a geometric interpretation of the problem which naturally leads the definition of the pre-image as a Karcher-mean [16] that interpolates between neighboring samples according to the diffusion distance. Previous pre-image methods were designed for Kernel PCA. Our motivation for using Diffusion Maps comes from the fact that the computed mapping captures the *intrinsic* geometry of the underlying manifold independently of the sampling. Therefore, the resulting Nyström extension (see Section 2.2) proves to be more “meaningful” far from the manifold and leads to quantitatively better pre-image estimations, even for very noisy input data. In the case of shape denoising, we compare our results to the work proposed by Dambreville [1] and for image denoising, to several denoising algorithms using Kernel PCA: [3], [2], [4]. Results on 3D shapes and 2D images are presented and demonstrate the superiority of our method.

The rest of the paper is organized as follows. Section 2 presents the Diffusion Maps framework and the out-of-sample extension. Section 3 introduces our pre-image methodology. Numerical experiments on real data are reported in section 4 and section 5 concludes.

## 2 Learning a Set of Shapes

Let  $\Gamma = \{s_1 \cdots s_p\}$  be  $p$  independent random points of a  $m$ -dimensional manifold  $\mathcal{M}$  locally sampled under some density  $q_{\mathcal{M}}(s)$  ( $m \ll p$ ). The manifold  $\mathcal{M}$  is assumed to be a smooth finite-dimensional sub-manifold embedded in a (potentially infinite-dimensional) space  $\mathbb{S}$ . The density  $q_{\mathcal{M}}(s)$  is unknown and might not be uniform. In this work, we consider more general spaces than the traditional Euclidean space  $\mathbb{R}^n$  and only assume that the input space  $\mathbb{S}$  is equipped with a distance  $d_{\mathbb{S}}$ .

### 2.1 Diffusion Maps

To extract the meaningful structure present in the training set  $\Gamma$ , classical manifold learning techniques minimize a quadratic distortion measure of the desired coordinates on the data, naturally leading to the eigenfunctions of Laplace-type operators as minimizers [8, 9]. Unfortunately, most unsupervised learning methods generate coordinates (the *embedding*) that combine the information of both the density  $q_{\mathcal{M}}$  and the geometry [9, 10, 18]. Diffusion Maps construct a discrete *density-independent* approximation of the Laplace-Beltrami operator  $\Delta_{\mathcal{M}}$  defined on  $\mathcal{M}$  and provide an embedding that captures the intrinsic geometry independently of the sampling density. We quickly review the construction of Diffusion Maps [8]. In a first step, we build a fully connected graph on the set  $\Gamma$  where each node corresponds to a sample  $\in \Gamma$ . Based on the distance ( $d_{\mathbb{S}}$ ) between samples, nodes are connected if their mutual distance is less or equal to  $\sigma$ , with  $\sigma$  being the median distance between all shapes. In order to build the normalized Laplacian matrix we use the diffusion kernel  $w(\cdot, \cdot)$

$$P_{i,j} = p(s_i, s_j) = \frac{w(s_i, s_j)}{g(s_i)}. \quad (1)$$

The diffusion kernel  $w(s_i, s_j)$  encodes the probability of transition between  $s_i$  and  $s_j$  and  $g(s_i)$  normalizes the quantity in (1) such that  $\sum_j p(s_i, s_j) = 1$ . Therefore, the quantity  $p(s_i, s_j)$  can be seen as the probability of a random walker to jump from  $s_i$  to  $s_j$  and  $P$  encodes a Markov Chain on  $\Gamma$ . The function  $g(s_i)$  measures the number of incident edges to the node corresponding to the shape  $s_i$ . If we introduce a time  $t$  and denote  $p_t$  the elements of  $P^t$  (the  $t^{\text{th}}$  power of  $P$ ), then  $p_t(s_i, s_j)$  corresponds to the probability of transition after  $t$  time steps. When  $t \rightarrow \infty$  the random walk converges to a unique stationary distribution  $\varphi_0$ . We have  $\varphi_0^T P = \varphi_0^T$ . Using a well known fact from spectral theory, Coifman [8] introduces the following eigen-decomposition of the kernel  $p_t$  :

$$p_t(s_i, s_j) = \sum_l \lambda_l^t \psi_l^t(s_i) \varphi_l^t(s_j), \tag{2}$$

where  $\{\lambda_l^t\}$  is the decreasing eigenspectrum of  $P^t$  and  $\{\varphi_l^t(s_j)\}$  respectively  $\{\psi_l^t(s_i)\}$  the corresponding biorthogonal left and right eigenvectors. They verify

$$\varphi_0(x) \psi_l(x) = \varphi_l(x). \tag{3}$$

Note that because of the fast decaying sequence of eigenvalues only a few terms need to be retained to approximate the probability  $p_t(., .)$  within a certain relative accuracy. Then the diffusion distance  $D_t(s_i, s_j)$  between two points  $s_i$  and  $s_j$  can be written as

$$D_t^2(s_i, s_j) = \sum_l \frac{(p_t(s_i, s_l) - p_t(s_j, s_l))^2}{\varphi_0(s_l)}. \tag{4}$$

This simple  $L^2$ -weighted distance between the two conditional probabilities  $p_t(s_i, .)$ ,  $p_t(s_j, .)$  defines a metric on the data that measures the amount of connectivity of the points  $s_i$  and  $s_j$  along paths of length  $t$ . In order to relate the diffusion distance we have to combine (2) and (4) and find with the biorthogonality relation between left and right eigenvectors(cf. [10]) that

$$D_t^2(s_i, s_j) = \sum_{l \geq 1} (\lambda_l^t \psi_l^t(s_i) - \lambda_l^t \psi_l^t(s_j))^2. \tag{5}$$

(since  $\psi_0$  is a constant vector, it is left out of the sum). Equation (5) shows that the right eigenvectors of  $P_t$  can be used to express the diffusion distance. To this end, we introduce the family of Diffusion Maps indexed by a time parameter  $t$

$$\Psi_t(s) = \begin{pmatrix} \lambda_0^t \psi_0^t(s) \\ \lambda_1^t \psi_1^t(s) \\ \vdots \end{pmatrix}$$

In the sequel we will omit the parameter  $t$  and assume it set to a fixed value [10]. From Equation (5), we can see that Diffusion aps generate a quasi-isometric mapping since the diffusion distance is approximately equal to the  $L^2$  metric in the new coordinate system when retaining the first  $m$  eigenvectors. Also note that methods like LLE or Laplacian Eigenmaps do not provide an explicit metric which is crucial for the contribution in this paper.

## 2.2 Out-of-Sample Extension

In general, the mapping  $\Psi$ , also referred to as an embedding, is only known over the training set. The extension of the mapping to new input points is of primary importance for kernel based methods whose success depend crucially on the “accuracy” of the extension. This problem, referred to as the *out-of-sample* problem, is often solved using the popular Nyström extension method [2, 19, 18]. Instead of recomputing the whole embedding, which can be costly for very large datasets because it involves a spectral decomposition, the problem is solved through a method borrowed from numerical analysis [20]. With this technique in hand and considering that every training sample verifies:

$$\forall s_j \in \Gamma \quad \forall l \in 1, \dots, p \quad \sum_{s_i \in \Gamma} p(s_j, s_i) \psi_l(y) = \lambda_l \psi_l(s_j),$$

the embedding of new data points located outside the set  $\Gamma$  can similarly be computed by a smooth extension  $\hat{\Psi}$  of  $\Psi$ :

$$\hat{\Psi} : \begin{cases} \mathbb{S} \rightarrow \mathbb{R}^p, s \mapsto (\hat{\psi}_1(s), \dots, \hat{\psi}_p(s)) \\ \forall l \in 1, \dots, p \quad \hat{\psi}_l(s) = \lambda_l \sum_{y \in \Gamma} p(s, y) \psi_l(y). \end{cases} \quad (6)$$

It is obvious that the extension depends on the data and recomputing the whole embedding with the new datum would yield a different embedding. But in general the approximation works well and is used throughout the literature. In addition, the reverse mapping from the feature space back to the input space is often required. After operations are performed in feature space (these operations necessitate the extension of the mapping), corresponding data points in input space often needs to be estimated. This problem, known as the *pre-image* problem, is the problem to be addressed in this paper. We now tackle the problem of pre-image computation using Diffusion Maps.

## 3 Pre-image as Karcher Means

We push the manifold interpretation and define the pre-image of  $\phi \in \mathbb{R}^p$  as the point  $s = \Psi_{|\mathcal{M}}^{-1}(\phi)$  in the manifold  $\mathcal{M}$  such that  $\Psi(s) = \phi$ . Although Diffusion Maps extract the global geometry of the training set and define a robust notion of proximity, they cannot permit the estimation of the manifold between training samples, i.e. the local geometry of the manifold is not provided. Following [21], we propose to approximate the manifold as the set of Karcher means [16] interpolating between correctly chosen subsets of  $m + 1$  sample points,  $m$  being the fixed dimension reduction parameter. Usually it is chosen by observing the eigenvalues of the eigenvectors. As mentioned in Section 2.1 only a few eigenvectors are needed to approximate well the diffusion distance. And the parameter  $m$  is exactly the number of eigenvectors retained. From a dimensionality reduction point of view this parameter corresponds to the degree of freedom in the data set but which cannot be computed automatically and therefore must be guessed. In [21], these subsets are the Delaunay simplices of a  $m$ -dimensional Delaunay triangulation of

the sample points. This limits in practice  $m$  to small values. Here, we simply exploit the Euclidean nature of the feature space: for a given  $\phi$ , we choose the interpolating subset as its  $m + 1$  nearest neighbors with respect to the diffusion distance  $D$ . We then define the pre-image  $s = \Psi_{|\mathcal{M}}^{-1}(\phi)$  as a Karcher mean that minimizes the mean-squared criterion:

$$s = \arg \min_{z \in \mathbb{S}} \|\Psi(z) - \phi\|^2 \tag{7}$$

### 3.1 Shape Interpolation Using Karcher Means

Given a set of neighboring points  $\mathcal{N} = \{s_1, \dots, s_{m+1}\}$  (i.e. neighboring for the diffusion distance  $D$ ), we assume that the manifold  $\mathcal{M}$  can be locally described (i.e. between neighboring samples) by a set of weighted-mean samples  $\{s_\Theta\}$  that verifies:

$$s_\Theta = \arg \min_{z \in \mathbb{S}} \sum_{1 \leq i \leq m+1} \theta_i d_{\mathbb{S}}(z, s_i)^2, \tag{8}$$

where  $d_{\mathbb{S}}$  is the distance in the input space and  $(\theta_i \geq 0, \sum_{i=1}^{m+1} \theta_i = 1)$ . The coefficients  $\Theta = \{\theta_1, \dots, \theta_{m+1}\}$  are the barycentric coefficients of the point  $s_\Theta$  with respect to its neighbors  $\mathcal{N}$  in  $\mathbb{S}$ . Proposed by Charpiat et al. [22], this model proved to give natural shape interpolations, compared to linear approximations. One classical choice is the area of the symmetric difference between the regions bounded by the two shapes:

$$d_{SD}(s_1, s_2) = \frac{1}{2} \int |\chi_{\Omega_1} - \chi_{\Omega_2}|, \tag{9}$$

where  $\chi_{\Omega_i}$  is the characteristic function of the interior of shape  $s_i$ . This distance was recently advocated by Solem in [23] to build geodesic paths between shapes. But the drawback is that this distance yields no unique geodesics. We proved this behavior analytically in the context of our method. But in the simulations we did not encounter any problems with the symmetric distance function. Another definition has been proposed [11, 24, 22], based on the representation of a curve in the plane, of a surface in 3D space, by its signed distance function. In this context, the distance between two shapes can be defined as the  $L^2$ -norm or the Sobolev  $W^{1,2}$ -norm of the difference between their signed distance functions. Let us recall that  $W^{1,2}(\Omega)$  is the space of square integrable functions over  $\Omega$  with square integrable derivatives:

$$d_{L^2}(s_1, s_2)^2 = \|\mathbb{D}_{s_1} - \mathbb{D}_{s_2}\|_{L^2(\Omega, \mathbb{R})}^2, \tag{10}$$

$$d_{W^{1,2}}(s_1, s_2)^2 = \|\mathbb{D}_{s_1} - \mathbb{D}_{s_2}\|_{L^2(\Omega, \mathbb{R})}^2 + \|\nabla \mathbb{D}_{s_1} - \nabla \mathbb{D}_{s_2}\|_{L^2(\Omega, \mathbb{R}^n)}^2, \tag{11}$$

where  $\mathbb{D}_{s_i}$  denotes the signed distance function of shape  $s_i$  ( $i = 1, 2$ ), and  $\nabla \mathbb{D}_{s_i}$  its gradient.

### 3.2 Pre-image and Manifold Interpolation

We propose to define the pre-image of a target point  $\phi$  in the feature space, as the point  $s_\Theta$  that minimizes the energy  $E_\Psi(s_\Theta) = \|\Psi(s_\Theta) - \phi\|^2$ ,  $s_\Theta$  being expressed a Karcher

mean for the neighborhood  $\mathcal{N}$  made of the  $m + 1$  samples of  $\Gamma$  which embedding are the  $m + 1$ -closest neighbors of  $\phi$  in the feature space equipped with  $D$ :

$$\Psi_{|\mathcal{M}}^{-1}(\phi) = \arg \min_{s_\Theta} \|\Psi(s_\Theta) - \phi\|^2, \quad (12)$$

where  $s_\Theta = \arg \min_{z \in \mathbb{S}} \sum_{1 \leq i \leq m+1} \theta_i d_{\mathbb{S}}(z, s_i)^2$

When the input space is some Euclidean space  $\mathbb{R}^n$  with its traditional  $L_2$ -norm, this indeed amounts to assuming that the manifold  $\mathcal{M}$  is piecewise-linear (i.e. linearly interpolated between neighboring training samples). For shapes, we will see that this yields natural pre-images. By simple extension, we define the projection of any new test sample  $s$  on the manifold  $\mathcal{M}$  by  $\Pi_{\mathcal{M}}(s) = \Psi_{|\mathcal{M}}^{-1}(\Psi(s))$ .

### 3.3 Implementation Issues

The pre-image  $\Psi_{|\mathcal{M}}^{-1}(\phi)$  is computed by gradient descent. Instead of optimizing over  $\Theta$ , we use a descent over  $s_\Theta$  itself (Equation 13), constraining it to remain a Karcher mean (Equation 8). This boils down to projecting the deformation field  $\nabla_s E_\psi$  onto the tangent space  $\mathbb{T}_{s_\Theta}^{\mathcal{M}}$  of  $\mathcal{M}$  at point  $s_\Theta$ . Note that to compute this tangent space, we are implicitly assuming that the space  $\mathbb{S}$  has a manifold structure, in particular that the tangent space  $\mathbb{T}_{s_\Theta}^{\mathbb{S}}$  of  $\mathbb{S}$  at location  $s_\Theta$  (i.e. the space of local deformations around  $s_\Theta$ ) is equipped with an inner product that we denote  $\langle \cdot | \cdot \rangle_{\mathbb{S}}$ .

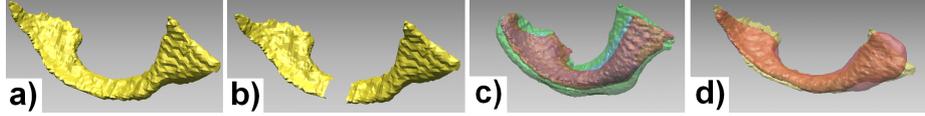
The optimality condition of Equation 8 is:

$$\forall \beta \in \mathbb{T}_{s_\Theta}^{\mathbb{S}}, \sum_{i=1}^{m+1} \theta_i d_i \langle \nabla_s d_i | \beta \rangle_{\mathbb{S}} = 0,$$

where we denote  $\mathcal{N} = \{s_1, \dots, s_{m+1}\}$  and  $d_i = d_{\mathbb{S}}(s_\Theta, s_i)$ . In order to recover the tangent space  $\mathbb{T}_{s_\Theta}^{\mathcal{M}}$  at  $s_\Theta$ , one needs to relate the  $m$ -independent modes of variations of the coefficient  $\Theta$  (remember that  $\sum_{i=1}^{m+1} \theta_i = 1$ ) with local deformation fields  $\mathbf{d}s_\Theta \in \mathbb{T}_{s_\Theta}^{\mathbb{S}}$ . To a small variation of the barycentric coefficients  $\Theta \rightarrow \Theta + \mathbf{d}\Theta$ , corresponds a small deformation of the sample  $s_\Theta \rightarrow s_\Theta + \mathbf{d}s_\Theta$ . Differentiating the optimality condition with respect to  $\Theta$  and  $s_\Theta$  provides the relation between  $\mathbf{d}\Theta$  and  $\mathbf{d}s_\Theta$ . For example, when the input space is taken to be the Euclidean space, i.e.  $\mathbb{S} = \mathbb{R}^n$ , we obviously obtain  $\mathbf{d}s_\Theta = \sum_1^{m+1} d\theta_i s_i$ . Remembering  $\sum_1^{m+1} d\theta_i = 0$  and fixing the  $d\theta_i$  appropriately, we can recover  $\mathbb{T}_{s_\Theta}^{\mathcal{M}}$ . Therefore we optimize for  $s_\Theta$  without explicitly computing  $\Theta$ . The gradient descent generates a family of samples  $s : \tau \in \mathbb{R}^+ \mapsto s(\tau) \in \mathcal{M}$  such that

$$s(0) = s_0, \quad \frac{ds}{d\tau} = -\mathbf{v}_{\mathcal{M}}(s_\tau),$$

with  $s_0 \in \mathcal{N}$  (in practice, the nearest neighbor of  $\phi$ ). The velocity field  $\mathbf{v}_{\mathcal{M}}(s_\tau)$  is the orthogonal projection of the deformation field  $\nabla_{s_\tau} E_\Psi = (\Psi(s_\tau) - \phi)^T \Lambda \Psi^T \nabla_{s_\tau} p_{s_\tau}$  onto the tangent space  $\mathbb{T}_{s_\tau}^{\mathcal{M}}$ . Here  $\Lambda$  is a diagonal matrix of eigenvalues and  $P s_i$  are the corresponding eigenvectors. Note that before projecting onto  $\mathbb{T}_{s_\tau}^{\mathcal{M}}$  we first orthogonalize the tangent space by using Gram-Schmidt. In the case of the  $L_2$ -norm the  $\Theta$ 's can be



**Fig. 2.** Interpolation using Karcher means for 39 three-dimensional sample shapes. From left to right: a) a new shape not in the given sample b) the same shape with an occlusion c) the 3 nearest neighbors of the corrupted shape according to the diffusion distance (in red, green and blue) d) the original shape (in yellow) and our interpolation (in red). See text for quantitative results.

easily recovered. When using a different distance function such as the symmetric difference or the Sobolev  $W^{1,2}$ -norm then one needs to solve additionally a system of linear equations in each step of the gradient descent.

## 4 Results

In order to validate the proposed method, we run several experiments on real and synthetic data. First, we test the Karcher mean interpolation with the reconstruction problem of occluded 3D medical shapes [1]. In a second experiment we validate the purpose of the projection of the gradient onto the tangent space. Finally, a third experiment demonstrates the superiority of our method for a standard denoising problem on images.

### 4.1 Remaining on the Manifold

To validate both the Karcher means modeling of the manifold and our projecting constraint (section 3.3), we generate a set of 200 synthetic shapes parameterized by an articulation angle and a scaling parameter (Fig. 3a). The corresponding embeddings are shown Fig. 3b. Choosing two distant shapes  $A$  and  $B$ , we compute a path  $s(\tau)$  from  $A$  to  $B$  be mean of a gradient descent starting from  $s(0) = A$  and minimizing  $d_S(s(\tau), B)$ . Fig. 3c and 3b show in red the intermediate shapes and the corresponding embeddings. In purple are shown the same path when projecting the gradient in order to remain on the manifold. Observe how the intermediate shapes look more like the original sample ones in that case. Note also that when remaining on  $\mathcal{M}$ , the interpolating path is almost a straight line with respect to the diffusion distance.

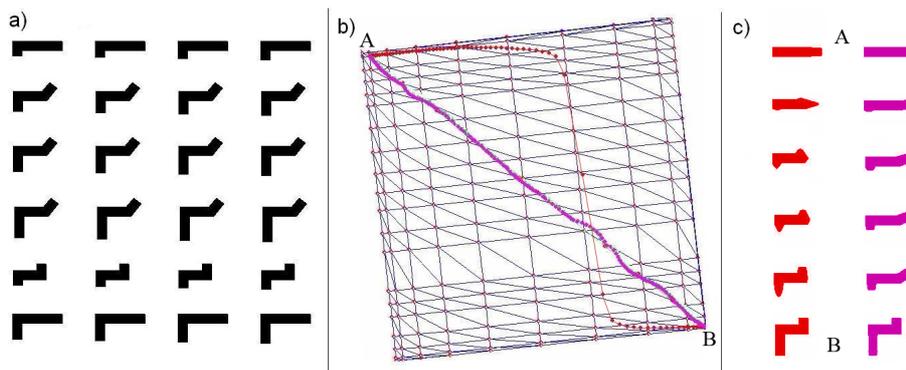
### 4.2 Projection and Manifold as Karcher Means

We here test the validity of using Karcher means as a manifold interpolation model. We consider the space of two-dimensional surfaces embedded in  $\mathbb{R}^3$ . For such a general space, many different definitions of the distance between two shapes have been proposed in the computer vision literature but there is no agreement on the correct way to measure shape similarity. In this work, we represent a surface  $s_i$  in the Euclidean embedding space  $\mathbb{R}^3$  by its signed distance function  $\mathbb{D}_{s_i}$ . In this context, we define the distance between two shapes to be the  $L_2$ -norm of the difference between their signed distance functions [11]:

$$d_S(s_1, s_2)^2 = \|\mathbb{D}_{s_1} - \mathbb{D}_{s_2}\|_{L^2}^2$$

**Table 1.** Average reconstruction error for a set of 9 noisy shapes

Avg err of shapes with occlusion	Nearest neighbors(NN)	Mean of NN	[1]	Our method
4.67	1.81	1.96	1.1	0.58



**Fig. 3.** Synthetic sample of 200 articulated and elongated shapes. From left to right: (a) a subset of the sample. (b) triangulated 2-dimensional embedding computed using Diffusion Maps and a gradient descent from an initial shape to a target one, without (red dots) and with (purple dots) remaining on the interpolated manifold. (c) Some shapes of the resulting evolution (left column: without projection, right column: with projection).

Note that, in order to define a distance between shapes that is invariant to rigid displacements (e.g. rotations and translations), we first align the shapes using their principal moments before computing distances. Note also that the proposed method is obviously not limited to a specific choice of distance [22, 17]. We use a dataset of 39 ventricles nuclei extracted from Magnetic Resonance Image (MRI). We learn a random subset of 30 shapes and corrupt the nine remaining shapes by an occlusion (Fig. 2a,b). In order to recover the original shapes we project the shapes onto the shape manifold with our method. We then compare the reconstruction results with the nearest neighbor, the mean of the  $m+1$  nearest neighbors and the method of Dambreville [1]. The parameters of this experiments is  $m = 2$ . In Figure 2-d one example of a reconstructed shape (red) is obtained from the  $m + 1$  nearest neighbors of  $s_\bullet$  (Fig. 2c). In order to quantitatively evaluate the projection, we define the reconstruction error as  $e(s) = d_S(s_\bullet, s)/\sigma$ , where  $s_\bullet$  is the original shape and  $s$  is the reconstructed shape. The occluded shape has an error of  $e(s_\bullet) = 4.35$ , while the nearest-neighbor has an error of 1.81. In Table 1 we see that our method is superior the one proposed by Dambreville [1].

### 4.3 Application: Denoising of Digits

To test the performance of our approach on the task of image denoising, we apply the algorithm on the USPS dataset of handwritten digits<sup>1</sup>. In a first experiment, we compare

<sup>1</sup> The USPS dataset is available from <http://www.kernel-machines.org>.

**Table 2.** Average PSNR (in dB) of the denoised images corrupted by different noise levels  $\sigma$ . Training sets consist in 60 samples (first 4 rows) and 200 samples (last 4 rows).

$\sigma^2$	[1]	[3]	[2]+ [1]	[2]+ [3]	[4]	Our method
0.25	8.50	15.71	10.17	16.18	14.01	17.71
0.45	9.05	13.87	9.98	15.42	13.91	17.52
0.65	9.78	13.10	9.58	13.60	13.89	17.38
0.85	9.06	12.58	8.61	13.91	13.87	17.32
0.25	9.35	16.08	11.97	16.21	15.27	17.95
0.45	9.64	15.70	10.18	15.98	14.85	17.85
0.65	9.41	13.97	10.26	15.85	14.13	17.79
0.85	9.24	13.06	10.25	15.07	14.07	17.75

our method to five state-of-the-art algorithms [1], [1]+ [2], [3], [3]+ [2] and [4]. For each of the ten digits, we form two training sets composed of randomly selected samples (60 and 200 respectively). The test set is composed of 40 images randomly selected and corrupted by some additive Gaussian noise at different noise levels. The process of denoising simply amounts to estimating the pre-images of the feature vectors given by the Nyström extension of the noisy samples. For all the methods, we take  $m = 8$  for the reduced dimension (number of eigenvectors for the kernel-PCA based methods). Table 2 shows a quantitative comparison based on the pixel-signal-to-noise ratio (PSNR). Our method outperforms visually (Fig. 1) and quantitatively other approaches. Interestingly, it is less sensitive to noise than other ones and yields good results even under heavy noise.

## 5 Conclusions and Future Work

In this paper, we focused on the pre-image problem. We provide a solution to the pre-image problem using Diffusion Maps. Following a manifold interpretation of the training set, we define the pre-image as a Karcher mean interpolation between neighboring samples with respect to the diffusion distance. Results on real world data, such as 3D shapes and noisy 2D images, demonstrate the superiority of our approach. In the continuation of this work several ideas may be exploited. In the perspective of working on complex shape spaces, our projection operator, defined from a manifold point-of-view, could be used in different tasks, such as segmentation with shape priors, interpolation and reconstruction of shapes, and manifold denoising. Interestingly, our approach is able to deal with manifolds of complex topology. In the context of manifold denoising this property can be useful. So far, none of the pre-image problems were tested when the training data itself contains heavy noise. We are currently investigating these directions.

## References

1. Dambreville, S., Rathi, Y., Tannenbaum, A.: Statistical shape analysis using kernel PCA. In: IS&T/SPIE Symposium on Electronic Imaging (2006)
2. Arias, P., Randall, G., Sapiro, G.: Connecting the out-of-sample and pre-image problems in kernel methods. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, June 18-23 (2007)

3. Kwok, J.T., Tsang, I.W.: The pre-image problem in kernel methods. *IEEE Transaction in Neural Network* 15(6), 1517–1525 (2004)
4. Carreira-Perpiñan, M.A., Lu, Z.: The Laplacian Eigenmaps Latent Variable Model. *JMLR W&P* 2, 59–66 (2007)
5. Roweis, S., Saul, L.: Nonlinear dimensionality reduction by locally linear embedding. *Science* 290, 2323–2326 (2000)
6. Tenenbaum, J.B., de Silva, V., Langford, J.C.: A global geometric framework for nonlinear dimensionality reduction. *Science* 290(5500), 2319–2323 (2000)
7. Belkin, M., Niyogi, P.: Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation* 15(6), 1373–1396 (2003)
8. Coifman, R., Lafon, S., Lee, A., Maggioni, M., Nadler, B., Warner, F., Zucker, S.: Geometric diffusions as a tool for harmonic analysis and structure definition of data: Diffusion maps. *PNAS* 102(21), 7426–7431 (2005)
9. Hein, M., Audibert, J.Y., von Luxburg, U.: From graphs to manifolds - weak and strong pointwise consistency of graph Laplacians. *Journal of Machine Learning Research*, ArXiv Preprint (forthcoming) (2006)
10. Lafon, S., Keller, Y., Coifman, R.R.: Data fusion and multicue data matching by diffusion maps. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 28(11), 1784–1797 (2006)
11. Leventon, M., Grimson, E., Faugeras, O.: Statistical shape influence in geodesic active contours. In: *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 316–323 (2000)
12. Cremers, D., Kohlberger, T., Schnörr, C.: Nonlinear shape statistics in mumford shah based segmentation. In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (eds.) *ECCV 2002*. LNCS, vol. 2351, pp. 93–108. Springer, Heidelberg (2002)
13. Lu, Z., Carreira-Perpinan, M., Sminchisescu, C.: People tracking with the laplacian eigenmaps latent variable model. In: Platt, J., Koller, D., Singer, Y., Roweis, S. (eds.) *Advances in Neural Information Processing Systems*, vol. 20, pp. 1705–1712. MIT Press, Cambridge (2008)
14. Pennec, X.: Intrinsic statistics on Riemannian manifolds: Basic tools for geometric measurements. *Journal of Mathematical Imaging and Vision* 25(1), 127–154 (2006); a preliminary appeared as INRIA RR-5093 (January 2004)
15. Davis, B., Fletcher, P., Bullitt, E., Joshi, S.: Population shape regression from random design data. In: *ICCV*, vol. 1 (2007)
16. Karcher, H.: Riemannian center of mass and mollifier smoothing. *Comm. Pure Appl. Math.* (30), 509–541 (1977)
17. Etyngier, P., Segonne, F., Keriven, R.: Shape priors using manifold learning techniques. In: *11th IEEE International Conference on Computer Vision*, Rio de Janeiro, Brazil (October 2007)
18. Lafon, S., Lee, A.B.: Diffusion maps and coarse-graining: a unified framework for dimensionality reduction, graph partitioning, and data set parameterization. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 28(9), 1393–1403 (2006)
19. Bengio, Y., Paiement, J.F., Vincent, P., Delalleau, O., Le Roux, N., Ouimet, M.: Out-of-sample extensions for lle, isomap, mds, eigenmaps, and spectral clustering. In: Thrun, S., Saul, L.K., Schölkopf, B. (eds.) *Advances in Neural Information Processing Systems*, vol. 16. MIT Press, Cambridge (2004)
20. Baker, C.T.H., Baker, C.T.H.: Numerical analysis of volterra functional and integral equations. In: Duff, I.S., Watson, G.A. (eds.) *The state of the art in numerical analysis*, pp. 193–222. University Press (1996)

21. Etyngier, P., Keriven, R., Segonne, F.: Projection onto a shape manifold for image segmentation with prior. In: 14th IEEE International Conference on Image Processing, San Antonio, Texas, US (September 2007)
22. Charpiat, G., Faugeras, O., Keriven, R.: Approximations of shape metrics and application to shape warping and empirical shape statistics. *Foundations of Computational Mathematics* 5(1), 1–58 (2005)
23. Solem, J.: Geodesic curves for analysis of continuous implicit shapes. In: International Conference on Pattern Recognition, vol. 1, pp. 43–46 (2006)
24. Rousson, M., Paragios, N.: Shape priors for level set representations. In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (eds.) ECCV 2002. LNCS, vol. 2351, pp. 78–92. Springer, Heidelberg (2002)