

Local High-order  
Regularization on Data  
Manifolds

James Tompkin, Kwang In Kim,  
Jan Kautz, Christian Theobalt

MPI-I-2014-4-001    January 2014



## Abstract

The common graph Laplacian regularizer is established in semi-supervised learning and spectral dimensionality reduction. However, as a first-order regularizer, it can lead to degenerate functions in high-dimensional manifolds. The iterated graph Laplacian enables high-order regularization, but it has a high computational complexity and so cannot be applied to large problems. We introduce a new regularizer which is globally high order and so does not suffer from the degeneracy of the graph Laplacian regularizer, but is also sparse for efficient computation in semi-supervised learning applications. We reduce computational complexity by building a local first-order approximation of the manifold as a surrogate geometry, and construct our high-order regularizer based on local derivative evaluations therein. Experiments on human body shape and pose analysis demonstrate the effectiveness and efficiency of our method.

## Contents

1	Tech Report	2
2	Appendix	12

# 1 Tech Report

# Local High-order Regularization on Data Manifolds

## Abstract

The common graph Laplacian regularizer is established in semi-supervised learning and spectral dimensionality reduction. However, as a first-order regularizer, it can lead to degenerate functions in high-dimensional manifolds. The iterated graph Laplacian enables high-order regularization, but it has a high computational complexity and so cannot be applied to large problems. We introduce a new regularizer which is globally high order and so does not suffer from the degeneracy of the graph Laplacian regularizer, but is also sparse for efficient computation in semi-supervised learning applications. We reduce computational complexity by building a local first-order approximation of the manifold as a surrogate geometry, and construct our high-order regularizer based on local derivative evaluations therein. Experiments on human body shape and pose analysis demonstrate the effectiveness and efficiency of our method.

## 1. Introduction

The graph Laplacian regularizer is established as one of the most popular regularizers for semi-supervised learning [5], spectral clustering [20, 14], and dimensionality reduction [3]. The underlying assumption for using the graph Laplacian regularizer is that data lie on a low-dimensional sub-manifold, and the object (e.g., a function) of interest should be regularized as defined on the manifold rather than as defined on the entire ambient space. By measuring local pairwise deviations of the function values in the ambient space, the graph Laplacian regularizer approximates the first-order variations on the manifold, thereby enabling us to regularize the function based on its first-order energy without having to know the manifold analytically.

Despite its solid theoretical background [4, 10] and success in many applications, the graph Laplacian regularizer has an important shortcoming that makes its usage less favorable on data lying in high-dimensional manifolds: as we will discuss, as a first-order regularizer, the null space of the graph Laplacian regularizer contains discontinuous functions on manifolds with dimensionality larger than 2 [16, 24].

Recently, Zhou and Belkin [24] proposed an iterated graph Laplacian approach that avoids this *degeneracy* and

enables regularization on high-dimensional manifolds. The price for the non-degeneracy and the resulting simplicity of the algorithm is high computational complexity: the iterated graph Laplacian regularizer is constructed by taking powers of the graph Laplacian matrix, which makes the original matrix denser and, accordingly, for large-scale problems (e.g.,  $O(100,000)$ ) it cannot be directly applied efficiently.

We propose an empirical regularizer which avoids degeneracy and leads to a sparse matrix. Our algorithm is based on the local linear approximation of the manifold: At each point, the corresponding neighborhood is projected onto its tangent space, where the high-order derivatives of the function are defined in this surrogate geometry. Instead of explicitly calculating high-order derivatives and measuring the corresponding complexity of the function, we measure its reproducing kernel Hilbert space (RKHS) norm. Similar to the graph Laplacian, its sparsity is explicitly controlled based on the local neighborhood structure. We present experimental results on human body shape and pose datasets, which show that our method is superior to graph Laplacian and iterated graph Laplacian techniques in terms of accuracy and computational complexity.

## 2. Problem statement

While our proposed regularizer can be used in clustering and dimensionality reduction, as with the graph Laplacian and iterated graph Laplacian regularizers, we focus on *semi-supervised learning* which enables us to compare numerically the performance of each algorithm.

For a set of data points  $\mathcal{X} = \{X_1, \dots, X_u\} \subset \mathbb{R}^n$  plus the corresponding labels  $\{Y_1, \dots, Y_l\} \subset \mathbb{R}$  for the first  $l$  points in  $\mathcal{X}$  where  $l \ll u$ , the goal of semi-supervised learning is to infer the labels of the remaining  $u - l$  data points in  $\mathcal{X}$ . Our approach is based on regularized empirical risk minimization:

$$\arg \min_{f: \mathbb{R}^n \rightarrow \mathbb{R}} \sum_{i=1}^l (Y_i - f(X_i))^2 + \lambda \mathcal{R}(f), \quad (1)$$

where  $\mathcal{R}(\cdot)$  is the regularization functional. Here, we use the standard squared loss function for simplicity, though our framework is applicable to any convex loss function. This problem can be solved either by reconstructing the underlying function  $f$  or by identifying its evaluation  $f|_{\mathcal{X}}$

on  $\mathcal{X}$ . In this paper, we focus on the second case, which is often called *transductive learning*.

Most semi-supervised learning algorithms can be characterized by how the unlabeled data points of  $\mathcal{X}$  are used to construct a corresponding regularizer  $\mathcal{R}(f|_{\mathcal{X}})$ . One of the best established regularizers is the graph Laplacian  $L$  [14]:

$$\mathcal{R}_L(\mathbf{f}) := \mathbf{f}^\top L \mathbf{f} = \sum_{i,j=1}^u [W]_{ij} (f_i - f_j)^2, \quad (2)$$

where  $f_i = f(X_i)$ ,  $\mathbf{f} := f|_{\mathcal{X}} = [f_1, \dots, f_u]^\top$ , and  $W$  is a non-negative input similarity matrix which is typically defined based on a Gaussian:

$$[W]_{ij} = \exp\left(-\frac{\|X_i - X_j\|^2}{b}\right). \quad (3)$$

One way of justifying the use of the graph Laplacian comes from its limit case behavior as  $u \rightarrow \infty$  and  $b \rightarrow 0$ : When the data  $\mathcal{X}$  is generated from an underlying manifold  $M$  with dimension  $m \leq n$ , i.e., the corresponding probability distribution  $P$  has support in  $M$ , the graph Laplacian converges to the Laplace-Beltrami operator  $\Delta$  on  $M$  [4, 10]. The Laplace-Beltrami operator can be used to measure the first-order variations of a continuously differentiable function  $f$  on  $M$ :

$$\|f\|_{\Delta}^2 := \int_M f(X) [\Delta f|_{\mathcal{X}}] dV(x) = \int_M \|\nabla f|_{\mathcal{X}}\|_g^2 dV(X), \quad (4)$$

where  $g$  is the *Riemannian metric*, and  $dV$  is the corresponding *natural volume element* [13] of  $M$ . The second equality is the result of Stokes' theorem. Accordingly, a graph Laplacian-based regularizer  $\mathcal{R}_L$  can be regarded as an empirical estimate of the first-order variation of  $f$  on  $M$  based on  $\mathcal{X}$ .

However, the convergence of the graph Laplacian  $L$  to the Laplace-Beltrami operator  $\Delta$  reveals an important shortcoming for it to be used as the standard regularizer for high-dimensional data: For high-dimensional manifolds ( $m > 1$ ), the null space of  $\Delta$  includes discontinuous functions on  $M$ . This is suggested by the *Sobolev embedding theorem* that states that, in general, any (semi-)norm induced by differential operators with order  $d \leq m/2$  will have discontinuous functions in its null space [18]. In particular, the norm  $\|\cdot\|_{\Delta}$  in Eq. 4 which measures the first-order variation has a null space consisting only of continuous functions (in particular, constant functions) when  $m = 1$  only. For  $m > 1$ , the null space of  $\Delta$  contains some discontinuous functions as a subset of  $L^2$  space which are equivalent almost everywhere to constant functions, except for the set of *measure zero* [7]. In other words, there are "spiky" functions  $f$ , e.g., Dirac delta functions, with norm  $\|f\|_{\Delta}^2 = 0$  (Fig. 1).

This is especially important in semi-supervised learning because we actively minimize the regularized risk of attaining a zero value by such a function (Eq. 1). While

this has been well-known in statistics, its effect on semi-supervised learning has only recently been analyzed by Nadler *et al.* [16]. They showed that, in the limit case (i.e.,  $u \rightarrow \infty$ ), where  $\mathcal{R}_L$  is used, indeed the null space of the empirical risk functional (Eq. 1) includes a function  $f$  which is zero everywhere except for the labeled data points  $\{X_1, \dots, X_l\}$ , where  $f$  agrees with the given labels, and no generalization is obtained.

In practice, due to the finite number of data points  $u$ , the learned function  $f$  (more precisely, its evaluation  $\mathbf{f}$  on  $\mathcal{X}$ ) is not a Dirac delta function exactly, but is a very steep, sheared spike which peaks at the labeled data points (Fig. 1). For discrete problems, e.g., classification, where only relative values of  $f$  are relevant, it is possible to normalize the output values based on the local distribution of  $f$  to soften such peaks, as exemplified in [22]. However, this technique is not applicable for learning continuous functions.

Zhou and Belkin [24] presented the first approach that explicitly prevents this degenerate case in semi-supervised learning. They proposed using powers of graph Laplacian (or *iterated graph Laplacian*) as a regularizer:

$$\mathcal{R}_{L^p}(\mathbf{f}) := \mathbf{f}^\top L^p \mathbf{f}, \quad (5)$$

with  $p > \frac{m}{2}$ . In the limit case as  $u \rightarrow \infty$ ,  $L^p$  converges to  $\Delta^p$ , which corresponds to the penalizer of (selected)  $\lceil \frac{p}{2} \rceil$ -th order variations in the context similar to Eq. 4 [24]:

$$\|f\|_{\Delta^p}^2 = \int_M f(X) [\Delta^p f|_{\mathcal{X}}] dV(X), \quad (6)$$

which is infinite when  $f$  is discontinuous. The ability to regularize over higher-order derivatives avoids the degenerate case of learning discontinuous functions.

One of the major limitations of iterated graph Laplacian is that, due to the density of the resulting matrix  $L^p$ , it cannot be directly applied to large-scale problems. For a non-iterated graph Laplacian, finding the minimizer of Eq. 1 with  $\mathcal{R}_L$  requires building and solving a linear system of size  $u \times u$ . Even for large-scale problems (e.g.,  $u \approx 10^5$ ), this is affordable since the corresponding weight matrix  $W$  can be well-approximated by a sparse matrix constructed from a  $k$ -nearest neighbor (NN) graph. However, in general, iterating  $L$  (taking powers  $L^p$ ) makes a sparse matrix denser. This is especially true when  $p$  is large, which is required for high-dimensional data, as suggested by the Sobolev embedding theorem. For instance, with  $u = 50,000$ , solving Eq. 1 with iterated graph Laplacian is  $15\times$  slower (Sec. 6) than the Laplacian case.

### 3. Local high-order regularization

Our goal is to build a new regularizer that shares the desirable properties of both penalizing discontinuous functions with  $L^p$  and being sparse in  $L$  for fast computation. To achieve this goal, we build a global regularization matrix  $G$

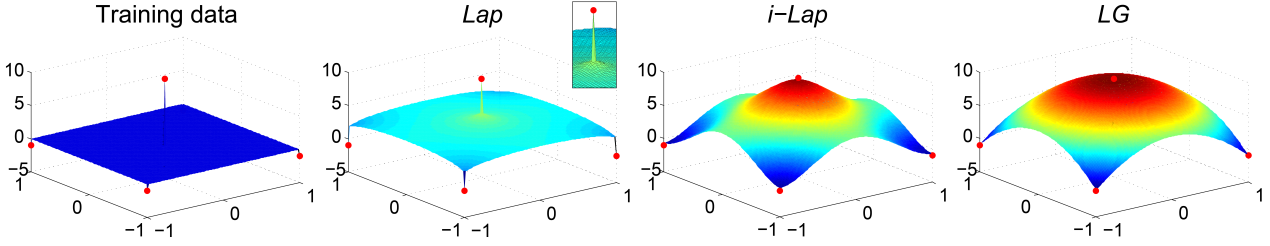


Figure 1. Example on 2D data. Section 5 contains details of this toy example; the surface in the training data plot is to help with visualization only, and no regularization has taken place. The *Lap* result largely fails to regularize, apart from points very near to the original training data. These spikes can be seen in the zoom inlay. The result of *i-Lap* looks *hyperbolic* because its null space includes polynomials. In this example, both *i-Lap* and *LG* are acceptable since they lead to smooth functions. Inspired by [24].

based on local regularizers evaluated at each point in  $\mathcal{X}$ .

First, we take a class of high-order manifold operators as regularizers by adopting the regularization framework of Yuille and Grzywacz [23]. These regularizers correspond to generalizations of Eq. 4:<sup>1</sup>

$$\|f\|_D^2 := \int_M \sum_{k=1}^{\infty} c_k |D^k f|_{\mathcal{X}}|^2 dV(X), \quad (7)$$

$$D^{2k} f = \Delta^k f, D^{2k+1} f = \nabla(\Delta^k f), \quad (8)$$

where  $c_k \geq 0$ ,  $|D^{2k} f|^2 := (D^{2k} f)^2$  for even cases, and  $|D^{2k+1} f|^2 := g(D^{2k+1} f, D^{2k+1} f)$  for odd cases.

For a known manifold with known metric and *Christoffel symbols* [13], the derivative operators in Eq. 8 are easy to calculate. However, in most practical applications, the manifold is not directly observed but is only indirectly observed as a point cloud of sampled data points  $\mathcal{X} \subset \mathbb{R}^n$ , where  $M$  is a ( $m$ -dimensional) sub-manifold of  $\mathbb{R}^n$ . Accordingly, direct calculation of Eq. 8 is infeasible.

**A local first-order approximation  $D_0$ .** We bypass this problem by using a local first-order approximation  $T_X(M)$  of manifold  $M$  at each point  $X$  ( $M_X$ ) in  $\mathbb{R}^n$  as a proxy geometry for  $M$  near  $X$ . Since  $T_X(M)$  is identified with  $\mathbb{R}^m$ , evaluating the derivative operators in Eq. 8 on  $X$  boils down to the calculation of the derivative operators in Euclidean geometry. In particular, evaluating the Laplace-Beltrami operator becomes the calculation of the Laplacian operator:

$$D_0^2 f|_X = \Delta_0 f|_X = \sum_{r=1}^m \partial_r^2 f|_X. \quad (9)$$

<sup>1</sup>As a special case, when  $c_p = 1$  and  $\{c_k\}_{k \neq p} = 0$ ,  $\|\cdot\|_D^2$  becomes  $\|\cdot\|_{\Delta_p}^2$  (Eq. 6). In general, different choices of differential operators are possible, e.g., Hessian, rather than the powers of  $\Delta$  and  $\nabla$ . This choice was motivated by the demonstrated empirical success of the resulting regularizer in many applications [23], and the computational efficiency as facilitated by the use of the corresponding Gaussian RKHS as discussed in Sec. 4.

Subscript 0 denotes operators defined on the proxy geometry, where  $\Delta_0[\cdot]|_X$  is the Laplacian defined at  $T_X(M)$ . Practically, the dimension of  $m$  is unknown and so is a hyperparameter.

With a manifold approximation, the next step is to construct approximations of Eq. 8 and Eq. 9 given  $\mathcal{X}$  and  $f|_{\mathcal{X}}$ . Suppose that for each data point  $X_i$ , the corresponding  $k$ -NN  $N_k(X_i) \subset \mathcal{X}$  are identified. First, we estimate the first-order approximation  $T_{X_i}(M)$  by performing principal component analysis on  $N_k(X_i)$  [6]: The representations  $\{\mathbf{x}_j\}_{j=1}^k$  of  $N_k(X_i)$  on  $T_{X_i}(M)$  are given as the first  $m$ -principal components of  $N_k(X_i)$ . Then, at  $X_i$ , the approximation of the Laplacian in Eq. 9 is obtained by fitting a smooth interpolation  $\varphi^i$  in  $(x)$  to  $\{f(X_j)\}_{j=1}^k$  and then extracting the trace of the resulting Hessian  $H\varphi^i$  of  $\varphi^i$ , which we denote as  $S^{(2)}(X_i)$ . The surrogate function  $\varphi^i$  can be a (constrained) second-order polynomial  $h^i$  (for  $\Delta$ ) or a Gaussian kernel interpolation  $g^i$  (for  $\Delta^k$ ,  $k > 0$ ):

$$h^i(\mathbf{x}) = f(X_i) + \sum_{r=1}^m [a^i]_r x^r + \sum_{r=1, s=r}^m [b^i]_{r,s} x^r x^s, \quad (10)$$

$$g^i(\mathbf{x}) = f(X_i) + \sum_{j=1}^k [\alpha^i]_j K(\mathbf{x}_j, \mathbf{x}), \quad (11)$$

where  $\mathbf{x} = [x^1, \dots, x^m]^\top$ , and

$$K(\mathbf{x}, \mathbf{x}') = \exp\left(-\frac{\|\mathbf{x} - \mathbf{x}'\|^2}{\sigma^2}\right). \quad (12)$$

The coefficients  $\{a^i, b^i\}$  and  $\{\alpha^i\}$  of  $h^i$  and  $g^i$ , respectively, are calculated as the standard least squares fits:

$$[a^i, b^i] = \arg \min_{w \in \mathbb{R}^{m+m(m+1)/2}} \sum_{j=1}^k \left(f(X_j) - h^i(\mathbf{x}_j)\right)^2, \quad (13)$$

$$\alpha^i = \arg \min_{a \in \mathbb{R}^k} \sum_{j=1}^k \left(f(X_j) - g^i(\mathbf{x}_j)\right)^2. \quad (14)$$



By combining these estimates of the local Laplacians and re-arranging the variables, one can construct a matrix  $B$  as a new regularizer on a point cloud  $\mathcal{X}$ :

$$\|f\|_{\Delta_0}^2 \approx \mathcal{R}_B(\mathbf{f}) = \mathbf{f}^\top B \mathbf{f} = \sum_{i=1}^u f(X_i) S^{(2)}(X_i). \quad (15)$$

To evaluate the squared Laplacian operator  $\Delta_0^2|_{X_i}$ , we calculate the corresponding fourth-order derivatives of  $\varphi$ . In the case when  $\varphi = g$ , the derivatives of  $\varphi$  of any order are easily calculated by noting that the derivative of a Gaussian function can be evaluated based on the original Gaussian and the combinations of Hermite polynomials [11]. The corresponding empirical regularizer  $\mathcal{R}_E$  based on a finite number of points  $\mathcal{X}$  can be constructed similarly to Eq. 15:

$$\mathcal{R}_E(\mathbf{f}) = \sum_{k=1}^{\infty} c_k \mathbf{f}^\top E^{(k)} \mathbf{f} := \sum_{i=1}^u \mathcal{S}_{X_i}(f), \quad (16)$$

where  $k$  indexes the order of derivatives,  $\mathcal{S}_{X_i}(f) = \sum_{k=1}^{\infty} c_k |S^{(k)}(X_i)|^2$ ,  $S^{(k)}(X_i)$  corresponds to an empirical approximation of  $D^k f|_{X_i}$ , and  $E^{(k)}(X_i)$  is the corresponding regularization matrix.

**Relation between  $D$  and  $D_0$ .** The regularizer  $\mathcal{R}_E$  depends on the local first-order approximation  $T_X(M)$  at each  $X$ . If the  $M$  is smoothly embedded in the ambient space  $\mathbb{R}^n$ , especially in the sense that the corresponding second fundamental form is bounded, then the approximation error is third-order: Let  $d_X := d_X(\cdot, \cdot)$  be the geodesic distance between two points on  $M$  in the neighborhood  $\mathcal{N}(X)$  of  $X$ ,<sup>2</sup> then the distance  $\tilde{d}_X$  between these points in the proxy geometry  $T_X(M)$  is related as [4, 10]

$$d_X = \tilde{d}_X + \mathcal{O}(\tilde{d}_X^3). \quad (17)$$

The use of local first-order approximations to a manifold is justified by its success in many applications (e.g., [19, 6]). We support this approximation further by noting that the corresponding orthonormal coordinates in  $T_X(M)$  can be regarded as approximations of *Riemannian normal coordinates* [12]. In a Riemannian normal coordinate chart centered at a point  $X$ , the manifold appears Euclidean up to second-order. Specifically, at  $X$ , the corresponding Riemannian metric  $g$  becomes Euclidean: the first order derivatives vanish, and evaluating the Laplace-Beltrami operator boils down to the calculation of the Laplacian in Euclidean space:

$$\Delta f|_X = \sum_{r,s=1}^m \frac{\partial_r (g^{rs} \sqrt{\det g} \partial_s f)}{\sqrt{\det g}} = \Delta_0 f|_X, \quad (18)$$

where  $\partial_r = \frac{\partial}{\partial x^r}$ ,  $\delta_s^r = \sum_t g^{rt} g_{ts}$ ,  $\delta_s^r: \delta_s^r = 1$  if  $r = s$  and 0, otherwise,  $g_{rs} = g(\partial_r, \partial_s)$ , and  $\det g$  is the determinant of the matrix evaluation  $\{g_{rs}\}$ . Using this setup, similarly to

<sup>2</sup>The *injectivity radius*  $\text{inj}(X)$  of  $X \in M$  is always positive [13]. Here, we assume that  $\mathcal{N}(X) \subset \text{inj}(X)$ .

the graph Laplacian  $L$  case, one can show the convergence of the matrix  $B$  (Eq. 15) to  $\Delta$  in the limit case as  $u \rightarrow \infty$ , the diameter  $\epsilon$  of  $N_k$  is controlled carefully:

**Definition 1 (Audibert and Tsybakov [1])** For given constants  $c_0, \epsilon_0 > 0$ , a Lebesgue measurable set  $A \subset \mathbb{R}^m$  is called  $(c_0, \epsilon_0)$ -regular if

$$\lambda[A \cap \mathcal{B}(\mathbf{x}, \epsilon)] \geq c_0 \lambda[\mathcal{B}(\mathbf{x}, \epsilon)], \quad \forall \epsilon \in [0, \epsilon_0], \forall \mathbf{x} \in A,$$

where  $\lambda[S]$  is the Lebesgue measure of  $S \subset \mathbb{R}^m$  [7]. We fix constants  $c_0, \epsilon_0 > 0$  and  $0 < \mu_{\min} < \mu_{\max} < \infty$  and a compact  $\mathcal{C} \subset \mathbb{R}^d$ . We say that the strong density assumption is satisfied if the distribution  $P$  is supported on a compact  $(c_0, \epsilon_0)$ -regular set  $A \subseteq \mathcal{C}$  and has a density  $\mu$  w.r.t.  $\lambda$  bounded above and below by between  $\mu_{\min}$  and  $\mu_{\max}$

$$\mu_{\min} \leq \mu(\mathbf{x}) \leq \mu_{\max}, \quad \forall \mathbf{x} \in A \text{ and } \mu(\mathbf{x}) = 0 \text{ otherwise.}$$

**Proposition 1** If Hessian  $Hf$  on  $M$  is Lipschitz continuous with the Lipschitz constant  $\gamma$ , and the natural volume element  $dV$  is bounded in the sense that the underlying probability distribution  $P$  satisfies strong density assumption, then there are constants  $C_1, C_2, \mu_0 > 0$  such that with probability larger than  $1 - (m^2 + 3m) \exp(-C_2 u \epsilon^m)$ :

$$|\text{tr}[Hh(\mathbf{x})] - \Delta f(X)|^2 \leq \frac{k}{u \epsilon^m} \frac{C_1 \epsilon^2 \gamma^2}{\mu_0}, \quad (19)$$

where  $\text{tr}[A]$  calculates the trace of  $A$ ,  $k = |\mathcal{X} \cap \mathcal{B}(X, \epsilon)|$ , and  $\mathcal{B}(X, \epsilon)$  is the  $\epsilon$ -neighborhood of  $X$  in coordinates, i.e.  $\mathcal{B}(X, \epsilon) := \{X' : \|\mathbf{x} - \mathbf{x}'\|_{T_X(M)} \leq \epsilon\}$ .

The proof of this convergence be found in the supplemental material. For simplicity of proof, we use the  $\epsilon$ -neighborhood  $\mathcal{B}(X, \epsilon)$  instead of  $k$ -NNs  $N_k(X)$ . It can be easily modified for the  $k$ -NN case (see supplemental material). Accordingly, in Eq. 19,  $\epsilon$  is the only parameter to be controlled to obtain the convergence. The role of  $\epsilon$  is similar to the width of the Laplacian *weight function* (Eq.3) in [4]: Roughly, decreasing  $\epsilon$  guarantees that the local surrogate function  $h$  is flexible enough to well-approximate  $f$ . However, it should not shrink too fast to ensure that there are sufficient data points  $k$  in  $\mathcal{B}(X, \epsilon)$  to prevent  $h$  from *overfitting* to  $f$ . This leads to the condition that  $\epsilon^m$ -shrink should be slower than  $u$ -increase, so that  $u \epsilon^m \rightarrow \infty$ . The number of neighborhoods  $k$  in Eq. 19, given as  $|\mathcal{B}(X, \epsilon) \cap \mathcal{X}|$ , is automatically controlled by sampling  $\mathcal{X}$  from  $P$ . This leads to  $\mathcal{O}(\frac{k}{u \epsilon^m}) = 1$  (see supplemental material) guaranteeing quadratic ( $\epsilon^2$ ) convergence. All other constants  $C_1, C_2, \mu_0$ , and  $\gamma$  are independent of  $u$ .

The strong density assumption is moderate. In particular, it holds for any compact manifold with a continuous distribution.

In general, the derivatives of the metric  $g$  with orders higher than 2 are non-vanishing even in normal coordinates. In this case, for instance,  $\Delta_0^2 f|_X$  deviates from  $\Delta^2 f|_X$  in

third-order:

$$\Delta^2 f|_X = \Delta_0^2 f|_X + \mathcal{D}^3(f|_X), \quad (20)$$

where  $\mathcal{D}^3(f|_X)$  contains selected derivatives of  $f$  at  $X$  up to third-order.<sup>3</sup>

However, since they agree at the highest (fourth) order,  $\Delta_0^2$  shares two important properties with  $\Delta^2$  which are precisely what leads to a *proper* regularizer for  $m < 4$ . When  $m < 4$ , and the metric  $g$  and the embedding  $\hat{i} : M \rightarrow \mathbb{R}^n$  are smooth:

1.  $c_2\Delta_0 + c_4\Delta_0^2$  with  $c_2, c_4 > 0$ , has the null space consisting of truly constant functions (i.e., excluding the degenerate functions which deviate from constant functions on sets of measure zero), and
2. The evaluation of the corresponding norm defined similarly to Eq. 4 is infinite for any discontinuous functions.

This property extends to general high-order cases: The approximation error of  $\Delta_0^k|_X$  to  $\Delta^k|_X$  is of order  $k - 1$  and, for a manifold with dimension  $m \geq 4$ , the regularizers  $\|\cdot\|_{D_0}^2$  that replaces  $D^k$  with  $D_0^k$  in  $\|\cdot\|_D^2$  (Eq. 7) with  $c^1, \dots, c^{\lfloor m/2+1 \rfloor} > 0$  share the same null space with  $\|\cdot\|_D^2$ . Furthermore, their evaluations on any discontinuous functions produce infinite value.

#### 4. Local Gaussian regularization

The regularization cost functional  $\mathcal{R}_E$  (Eq. 16) has both the desired properties of being a high-order regularizer and of leading to a sparse system. However, evaluating it requires explicitly calculating the powers of the Laplacian evaluation  $\Delta_0^k f|_{X_i}$  at each point  $X_i \in \mathcal{X}$  and for each non-zero coefficient  $c_k$ . This is not only tedious but also numerically unstable since, in practice, the corresponding high-order derivatives are estimated by fitting a function  $\varphi^i$  to only a small number ( $k$ ) of data points  $N_k(X_i)$ : fitting a high-order polynomial (as an extension of  $h^i$  in Eq. 11) is very unstable in general. While this can be resolved with smooth Gaussian interpolation i.e.  $\varphi^i = g^i$ , due to the existence of high-order polynomials contained in the derivatives of  $g^i$  (Eq. 11), the resulting derivative estimates can still be unstable, i.e., perturbed significantly with respect to slight variations of  $f$ .

We focus on a special case of the regularization functional  $\mathcal{R}_E$ , with a specific choice of derivative operator contribution  $\{c_k\}$ , which enables us to bypass the explicit evaluation

<sup>3</sup>This can be easily verified by expanding the derivatives in normal coordinates at  $X$ :

$$\Delta^2 f = \sum_{i,j,r,s=1}^m \left( \partial_i \partial_j [g^{rs} \partial_r \partial_s f] + \partial_i \partial_j [\partial_r [\partial g^{rs}] \partial_s f] + \frac{1}{2} \partial_i \partial_j \left[ g^{rs} \sum_{t,u=1}^m g^{tu} \partial_r [\partial g^{tu}] \partial_s f \right] \right).$$

of individual derivatives  $D^k$  while retaining the desired properties of being a sparse, robust, and high-order regularizer.

First, the stability problem in evaluating derivatives can be addressed by taking integral averages of derivative evaluations ( $D^k f$ ; Eq. 8) and the corresponding magnitude  $|D^k f|$  within a neighborhood  $\mathcal{U}(X_i)$  of  $X_i$ , rather than their point evaluations at  $X_i$ . For instance, for derivative operators of even powers, instead of  $|D_0^{2k} f|_{X_i}|$  (Eq. 7), we use:

$$|\tilde{D}_0^{2k} f|_{X_i}| = \frac{1}{\text{vol}(\mathcal{U}(X_i))} \int_{\mathcal{U}(X_i)} [\Delta_0^k \varphi^i|_{\mathbf{x}}|^2] d\mathbf{x}, \quad (21)$$

where  $\text{vol}(A)$  measures the volume of  $A \subset T_{X_i}(M)$ , which is a fixed constant given  $M$ .

This still requires explicit calculation of derivatives. However, for the special case of Eq. 7 where the coefficients  $\{c_k\}$  are given as:

$$c_k = \frac{\sigma^{2k}}{k!2^k}, \quad (22)$$

with  $\sigma^2$  as defined in (11) we can efficiently calculate an approximation: First, the *local energy* of  $\varphi^i = g^i$  over  $T_{X_i}$  defined as

$$\|g^i\|_D^2 := \sum_{k=1}^{\infty} c_k \int_{T_{X_i}(M)} |D^k g^i|_{\mathbf{x}}|^2 d\mathbf{x} = \|g^i\|_K^2, \quad (23)$$

can be analytically evaluated as the corresponding Gaussian reproducing kernel Hilbert space (RKHS) norm  $\|\cdot\|_K$ : We obtain the second equality by solving the Euler-Lagrange equation that adopts the energy in Eq. 7 with  $c_k$  given in Eq. 22 [23].

Second, we note that, for large  $u$ , the local energy (Eq. 23) well approximates the sum of local stabilized derivations (Eq. 21). For a Gaussian function  $K(\mathbf{x}_j, \cdot)$ , its value and derivatives decrease rapidly as the corresponding points of evaluation deviate from center  $X_j$  (depending on its *width*  $\sigma^2$ ). Accordingly, its support is *effectively* limited within a neighborhood  $\mathcal{U}'(X_j)$ . Since  $D^k g^i$  is a kernel expansion of  $N_k(X_i)$ , its support is limited to a larger neighborhood  $\mathcal{N}(X_i)$  of  $X_i$  that encompasses  $\{\mathcal{U}'(X_j), \forall X_j \in N_k(X_i)\}$ . Then, we set  $\mathcal{U}(X_i)$  by  $\mathcal{N}(X_i)$  and obtain the local energy  $\|g^i\|_D^2$  as a replacement of the integrand in (7).

In general, for given  $\mathcal{U}(X_i)$ , this approximation becomes more accurate as  $\sigma^2$  and  $N_k(X_i)$  decrease to zero, which is the case as  $u \rightarrow \infty$  (see accompanying supplemental material). However, for practical applications, we do not tune  $\sigma^2$  or  $N_k(X_i)$  to minimize error or to achieve a desired level of accuracy since explicitly calculating the corresponding error is tedious (see Appendix). More importantly, having too small  $\sigma^2$  or  $N_k(X_i)$  for finite  $u$  will lead to a bad interpolation function: a Gaussian kernel interpolation with small  $\sigma^2$  may lead to a highly non-linear function  $g^i$  that overfits to  $\{f(X_j)\}_{j=1}^k$ . While we propose setting  $\sigma^2$  and  $N_k(X_i)$  as decreasing functions with respect to  $u$  so that

the approximation becomes exact as  $u \rightarrow \infty$ , for practical applications with fixed  $u$  (including our experiments), we implicitly determine the diameter of  $N_k(X_i)$  based on the selected  $k$ -NN, and regard  $k$  and  $\sigma^2$  as hyper-parameters. As described in Sec. 6,  $\sigma^2$  is actually adaptively determined based on  $N_k(X_i)$  and accordingly only  $N_k(X_i)$  is tuned.

Now, we build a new regularizer  $\mathcal{R}_G$  as a combination of local regularizers on  $\varphi^i - f(X_i)$  for  $i = 1, \dots, u$ :

$$\mathcal{R}_G(\mathbf{f}) = \sum_{i=1, \dots, u} \mathbf{f}^i \mathbf{G}^i \mathbf{f}^i \quad (24)$$

with

$$\mathbf{f}^i \mathbf{G}^i \mathbf{f}^i = \|f(X_i) - \varphi^i(\cdot)\|_K^2 \quad (25)$$

$$= \mathbf{f}^i \mathbf{T} (I - \mathbf{1}\mathbf{1}^i) \mathbf{T} (\mathbf{K}^i)^+ (I - \mathbf{1}\mathbf{1}^i) \mathbf{f}^i, \quad (26)$$

where  $[\mathbf{K}]_{lm} = K(\mathbf{x}_l, \mathbf{x}_m)$ ,  $\mathbf{f}^i = [f(X_1), \dots, f(X_k)]^\top$ ,  $\mathbf{K}^+$  is the Moore-Penrose pseudoinverse of  $\mathbf{K}$ , and  $\mathbf{1}\mathbf{1}^i$  is an indicator matrix whose element is zero except for the  $l(i)$ -th column that consists of ones with  $l(i)$  being the index of  $X_i$  in  $N_k(X_i)$ .

## 5. Augmenting null spaces

Our local Gaussian regularizer completely eliminates the possibility of generating degenerate functions and so provides a valid regularization on high-dimensional manifolds. Further, it is designed as a combination of *local* regularizers (Eq. 24) and so is tailored to incorporate a priori knowledge of the local behavior of functions. In particular, it is easy to tune the regularizer such that it does not penalize functions with desirable properties (i.e., to augment the null space of the regularizer so that it contains those functions). One good choice for  $\mathbf{f}$  are geodesic functions: both Donoho and Grimes [6] and Kim et al. [12] have demonstrated that *geodesic functions*, which are linear along geodesics, i.e., nothing more than linear functions in Euclidean space, are preferred over other functions since they correspond to the most *natural* parametrization of the underlying data.

The geodesic functions are completely characterized by their local behavior. In particular, in the Riemannian normal coordinates, they are locally linear functions. Accordingly, we can easily add geodesic functions to the null space of the global regularizer  $\mathcal{R}_G(\mathbf{f})$  by including linear functions in the null space of the local regularizers (Eq. 26): We fit a linear function to  $\mathbf{f}^i$  and *subtract* the resulting function from  $\mathbf{f}^i$  before we fit the non-linear function (Eq. 11). This can be easily incorporated into new local regularization matrices:

$$(\mathbf{G}')^i = \|f(X_i) - \varphi_L^i(\cdot) - \varphi^i(\cdot)\|_K^2 \quad (27)$$

$$= (\mathbf{L}^i)^\top (\mathbf{K}^i)^+ \mathbf{L}^i, \quad (28)$$

where  $\varphi_L^i(\cdot)$  is the linear regressor fitting  $\mathbf{f}^i$  in normal coordinates (i.e.,  $\varphi_L^i(\mathbf{x}) = (\Phi_L^i)^+ (I - \mathbf{1}\mathbf{1}^i) \mathbf{f}^i \mathbf{x}$ ),  $\Phi_L^i \in \mathbb{R}^{k \times m}$  is the design matrix whose rows correspond to the normal

---

**Algorithm 1:** The construction of the regularization functional  $\mathcal{R}_{G'}$  from a point cloud  $\mathcal{X}$ .

---

**Input:**  $\mathcal{X} = \{X_1, \dots, X_u\}$ , manifold dimension  $n$ ,  $k$ .

**Output:**  $G'$ .

Initialization: Find  $k$  nearest neighbors, e.g., build KD-tree;

**for**  $i = 1, \dots, u$  **do**

    Construct the local approximation  $M$  at  $X_i$  using  $n$ -dimensional PCA of  $N_k(X_i)$ ;

    Calculate the local regularization matrix  $\mathbf{G}^i$  for  $N_k(X_i)$  in the PCA representation:

$(\mathbf{G}')^i = (\mathbf{L}^i)^\top (\mathbf{K}^i)^+ \mathbf{L}^i$  (Eqs. 28 and 29);

**end**

Re-arrange  $\{(\mathbf{G}')^i\}$  according to the indices of  $\{\mathbf{f}^i\}$  in  $\mathbf{f}$  to construct matrix  $G'$  s.t.  $\mathbf{f}^\top G' \mathbf{f} = \mathcal{R}_{G'}(\mathbf{f})$ ;

---

coordinate values of  $N_k(X_i)$ , and

$$\mathbf{L}^i = I - \mathbf{1}\mathbf{1}^i - \Phi_L^i (\Phi_L^i)^\top (I - \mathbf{1}\mathbf{1}^i). \quad (29)$$

The new regularization functional  $\mathcal{R}_{G'}$ , in which  $\{(\mathbf{G}')^i\}$  replaces  $\{\mathbf{G}^i\}$ , has a richer null space: a one-dimensional space of constant functions plus an  $m$ -dimensional space of geodesic functions. This null space should not be confused with the *too large* null space of the original graph Laplacian regularizer. The null space of our updated local Gaussian regularizer does not include any degenerate functions.

While this setup does not cause any noticeable increase of computational complexity, in our preliminary MoCap experiments (see Sec. 6), this reduced error rates by around 3%. Accordingly, throughout the entire experiments, we use this new local Gaussian regularizer.

$\mathcal{R}_{G'}$  construction pseudocode is in Algorithm 1.

## 6. Experiments

To demonstrate our algorithm performance, we consider examples of estimating continuous values in human body shape and pose analysis: the MoCap database [2] of optical motion capture data and the CAESAR human body database [17]. For comparison, we performed experiments with existing graph Laplacian (*Lap*) [14, 3] and iterated graph Laplacian (*i-Lap*) [24] regularizers.

**Toy example.** We uniformly sample 10,000 data points in  $[-1, 1] \times [-1, 1]$ . Five points (four corners and center) were assigned labels in  $\{-1, 10\}$  (red dots in Fig. 1). While the original graph Laplacian (*Lap*) produces a “spiky” function, the iterated graph Laplacian (*i-Lap*) and our regularizer (*LG*) produced smooth functions, which demonstrate the importance of high-order regularization.

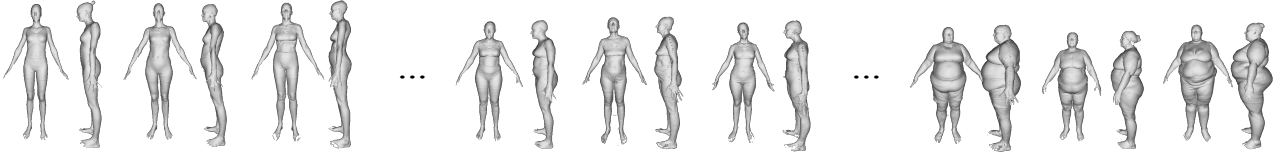


Figure 2. The CAESAR database contains 4,258 3D scans of human beings, along with ground-truth body measurements taken with calipers. Here, we see variation in female shape across the database.

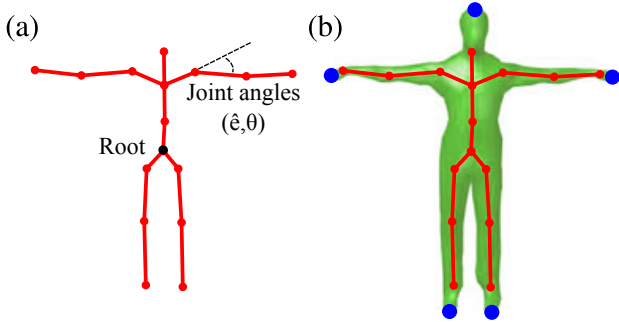


Figure 3. (a) Skeletal kinematic chain. (b) End effectors (blue) recovered from a geometric model fit to the skeleton. Each joint angle is in angle-axis form, with axis  $\hat{e}$  and angle  $\theta$ .

**MoCap database.** This contains 50,000 entries describing human body poses captured with an optical marker-based system [2]. For each *pose* entry, inverse kinematics is applied to recover skeletal joint angles represented as axis-angle  $(\hat{e}, \theta)$ . A body model comprising a surface mesh consisting of 6,449 vertices is deformed via surface skinning by embedding this skeleton of 62 joints, leading to 42 degrees of freedom parameterized by the joint angles. The locations of end effectors (left/right hand, left/right foot, and head) were separately recorded from the surface mesh model. These constitute a 15 ( $5 \times 3$ )-dimensional coarse, mid-level representation (Figure 3). The task is to estimate the 42-dimensional joint angles from the mid-level representation. This is useful for retrieval and indexing of motion data, e.g., for motion capture with motion priors of similar poses [2], fast MoCap data indexing in authoring tools [15], or synthesis of motions from sparse sensor data with pose priors [21].

We randomly chose 100 labels, with the remaining data points used as unlabeled examples. The experiment was repeated 10 times with different sets of labeled examples and the results were averaged (Table 1). We also show the corresponding results measured in the 186 ( $62 \times 3$ )-dimensional joint location space that is restored by applying forward kinematics. Both in terms of joint angle and position error, we outperform the competing methods.

**CAESAR database.** This contains 4,258 3D scans of human beings, along with ground-truth measurements of their bodies obtained with calipers (Fig. 2). Detailed description

Table 1. Mean L2-reconstruction error on the MoCap dataset.

Algorithm	<i>Lap</i>	<i>i-Lap</i>	<i>LG</i>
Joint angles error	1.62	1.24	<b>1.16</b>
Joint locations error	1.22	0.72	<b>0.50</b>

and example usages of this dataset can be found in [9]. With the technique of Hasler et al. [8], we fit a statistical body model to each of the scans, which is able to represent body variations such as height, hip and belly girth, limb length, and so on. Each body scan is represented as a vector in 20-dimensional feature space spanned by a linear shape basis.

Table 2 shows absolute error in semi-supervised learning performance when comparing the three regularizers, over different numbers of labeled items. Each experiment was repeated 10 times and averaged. In most cases, our approach improves performance, reducing errors by 3 – 388% over *Lap* (mean 86%), and 1 – 194% over *i-Lap* (mean 32%). The worse performance of *LG* over *i-Lap* for some cases is caused by over-fitting in cross-validation.

**Parameters.** There are four hyper-parameters in our algorithm: the number ( $k$ ) of nearest neighbors, the dimensionality ( $m$ ) of the manifold, the regularization parameter ( $\lambda$ ), and the local scale parameter ( $\sigma$ ; see Eq. 22). In preliminary experiments, the performance of our algorithm varied significantly with respect to the first three parameters, while it was rather robust to  $\sigma$  variations. We decide  $\sigma$  adaptively for each point  $X_i$ , at 0.1 times the mean distance between  $X_i$  and the elements of  $N_k(X_i)$  while the remaining three hyper-parameters were optimized by 5-fold cross-validation (CV) where, in each run, a subset of labeled points were left out while all unlabeled data points are kept. There are three and four hyper-parameters for *Lap* and *i-Lap*, respectively:  $\lambda$ ,  $k$ , and the parameter  $b$  for building the graph Laplacian (Eq. 3) for *Lap* and the iteration parameter  $p$  for *i-Lap* (Eq. 5). These parameters were tuned in the same way as for *LG*. Across Table 2,  $k$  varied from 20 to 40,  $m$  from 10 to 17,  $\lambda$  from  $10e^{-8}$  to  $10e^{-5}$ ,  $b$  from 5 to 300, and  $p$  from 1 to 4.

**Computation complexity and time.** For each algorithm, this depends on the number of data points  $u$ , the number of nearest neighbors  $k$ , and the number of non-zeros entries



Table 2. Mean absolute error for estimating 6 ground truth parameters from the CAESAR dataset. Bold face marks the best results. The *Deviation from mean* replaces the evaluation of each  $f(X_i)$  with the mean of each output variable (calculated from the entire data set). This presents an idea of the difficulty of the estimation problem for each parameter.

# Labels	Algorithm	Age	Arm length	Shoulder breadth	Weight	Sit height	Foot length
	<i>Deviation from mean</i>	10.89	35.98	36.13	13.94	39.50	15.57
20	<i>Lap</i>	<b>10.89</b>	30.23	32.69	12.80	32.58	13.80
	<i>i-Lap</i>	12.46	19.54	25.34	6.30	20.54	10.30
	<i>LG</i>	12.55	<b>17.92</b>	<b>20.64</b>	<b>3.17</b>	<b>19.31</b>	<b>9.87</b>
50	<i>Lap</i>	10.79	24.28	28.88	10.99	26.05	11.14
	<i>i-Lap</i>	<b>10.61</b>	17.43	21.14	6.62	18.39	<b>8.20</b>
	<i>LG</i>	11.03	<b>16.30</b>	<b>16.15</b>	<b>2.25</b>	<b>16.49</b>	8.34
100	<i>Lap</i>	10.64	20.62	26.00	9.60	21.72	9.46
	<i>i-Lap</i>	10.21	16.97	19.33	5.08	17.65	<b>7.99</b>
	<i>LG</i>	<b>9.85</b>	<b>15.07</b>	<b>15.39</b>	<b>1.98</b>	<b>15.59</b>	8.05
200	<i>Lap</i>	10.45	18.23	23.07	8.09	18.99	8.38
	<i>i-Lap</i>	9.99	16.49	17.56	4.11	17.25	7.81
	<i>LG</i>	<b>9.40</b>	<b>13.96</b>	<b>14.93</b>	<b>1.77</b>	<b>12.42</b>	<b>7.76</b>
500	<i>Lap</i>	10.00	16.44	19.39	6.02	17.31	7.75
	<i>i-Lap</i>	9.52	15.62	15.84	2.93	16.65	7.59
	<i>LG</i>	<b>8.93</b>	<b>13.42</b>	<b>14.53</b>	<b>1.60</b>	<b>11.94</b>	<b>7.54</b>

of the resulting regularization matrix that lies in-between  $O(uk)$  and  $O(uk^2)$ , depending on the well-behavedness of neighborhoods (where  $O(uk^2)$  corresponds to random neighbors). The most time-consuming component of each algorithm is solving the corresponding system.

For the MoCap dataset, with  $u = 50,000$ ,  $k = 20$ , and  $p = 4$  for *i-Lap*, it took 30, 50, and 40 seconds for *Lap*, *i-Lap*, and *LG* to build the regularization matrices, respectively. The corresponding sparsity, defined as the number of nonzero entries divided by the number of all entries in the regularization matrix, is 0.0005, 0.0400, and 0.0017 for *Lap*, *i-Lap*, and *LG*, respectively. This resulted in the run-times for solving the systems of roughly 50, 720, and 120 seconds, respectively, on an Intel Xeon 3GHz CPU in MATLAB. For the CAESAR dataset, with  $u = 4,258$ , run-times were only a few seconds. The improvement in computation time for large sets, coupled with the accuracy improvements demonstrated, makes our new regularizer a good alternative to *Lap* and *i-Lap*. Our code will be made available on the web.

## 7. Discussion

Local first-order approximation approaches, like ours, are supported by their success in manifold learning and regularization [19, 6]. However, local first-order approximations result in the corresponding derivatives being exact up to second order, but at third order and higher, the derivatives may deviate from the underlying covariant derivatives. Nevertheless, since the highest-order terms agree, calculating

the Euclidean derivatives therein enables us to completely eliminate the possibility of generating degenerate functions.

Furthermore, the number of hyper-parameters to be tuned (the other parameter  $\sigma^k$  is adaptively decided) is the same as for classical graph Laplacian and is one smaller than for iterated graph Laplacian. Combined with the observed empirical performance of our algorithm, and the computationally efficient regularization, this supports its usage.

Our local Gaussian interpolation varies  $\sigma^k$  with the local neighborhood size  $N_k(X)$  (instead of making it constant per dataset), which desires rigorous limit case behavior analysis. Further future work should address the theoretical analysis of our regularizer (e.g., error bound), and the possible benefit to spectral clustering and dimensionality reduction.

## 8. Conclusion

We have presented the local Gaussian regularizer: a new high-order regularization framework on data manifolds. Our algorithm does not suffer from the degeneracy of graph Laplacian-based regularizers. Further, it leads to a sparse regularization matrix, thereby facilitating application to large-scale datasets. Experiments on human body shape and pose analysis demonstrate the improved accuracy and faster execution time of our new algorithm.

## References

- [1] J.-Y. Audibert and A. B. Tsybakov. Fast learning rates for plug-in classifiers. *The Annals of Statistics*, 35(2):608–633, 2007. 4
- [2] A. Baak, M. Müller, G. Bharaj, H.-P. Seidel, and C. Theobalt. A data-driven approach for real-time full body pose reconstruction from a depth camera. In *Proc. ICCV*, pages 1092–1099, 2011. 6, 7
- [3] M. Belkin and P. Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation*, 15(6):1373–1396, 2003. 1, 6
- [4] M. Belkin and P. Niyogi. Towards a theoretical foundation for Laplacian-based manifold methods. *Journal of Computer and System Sciences*, 74(8):1289–1308, 2005. 1, 2, 4
- [5] O. Chapelle, B. Schölkopf, and A. Zien. *Semi-Supervised Learning*. MIT Press, Cambridge, MA, 2006. 1
- [6] D. L. Donoho and C. Grimes. Hessian eigenmaps: locally linear embedding techniques for high-dimensional data. *Proc. of the National Academy of Sciences*, 100(10):5591–5596, 2003. 3, 4, 6, 8
- [7] R. M. Dudley. *Real Analysis and Probability*. Cambridge University Press, Cambridge, 2nd edition, 2002. 2, 4
- [8] N. Hasler, C. Stoll, M. Sunkel, B. Rosenhahn, and H.-P. Seidel. A statistical model of human pose and body shape. *Computer Graphics Forum*, 28(2):337–346, 2009. 7
- [9] S. Hauberg, O. Freifeld, and M. J. Black. A geometric take on metric learning. In *NIPS*, pages 2033–2041, 2012. 7
- [10] M. Hein, J.-Y. Audibert, and U. von Luxburg. From graphs to manifolds - weak and strong pointwise consistency of graph Laplacians. In *Proc. COLT*, pages 470–485, 2005. 1, 2, 4
- [11] M. Kara. An analytical expression for arbitrary derivatives of Gaussian functions  $\exp(ax^2)$ . *Internal Journal of Physical Sciences*, 4(4):247–249, 2009. 4
- [12] K. I. Kim, F. Steinke, and M. Hein. Semi-supervised regression using Hessian energy with an application to semi-supervised dimensionality reduction. In *NIPS*, pages 979–987, 2010. 4, 6
- [13] J. M. Lee. *Riemannian Manifolds- An Introduction to Curvature*. Springer, New York, 1997. 2, 3, 4
- [14] U. Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17(4):395–416, 2007. 1, 2, 6
- [15] M. Müller, T. Röder, and M. Clausen. Efficient content-based retrieval of motion capture data. *ACM Trans. Graphics (Proc. SIGGRAPH)*, 24(3):677–685, 2005. 7
- [16] B. Nadler, N. Srebro, and X. Zhou. Statistical analysis of semi-supervised learning: the limit of infinite unlabelled data. In *NIPS*, pages 1330–1338, 2009. 1, 2
- [17] K. Robinette, H. Daanen, and E. Paquet. The CAESAR project: a 3-D surface anthropometry survey. In *Proc. 3-D Digital Imaging and Modeling*, pages 380–386, 1999. 6
- [18] S. Rosenberg. *The Laplacian on a Riemannian Manifold: An Introduction to Analysis on Manifolds*. Cambridge University Press, Cambridge, 1997. 2
- [19] S. Roweis and L. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, pages 2323–2326, 2000. 4, 8
- [20] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE TPAMI*, 22(8):888–905, 2000. 1
- [21] J. Tautges, A. Zinke, B. Krüger, J. Baumann, A. Weber, T. Helten, M. Müller, H.-P. Seidel, and B. Eberhardt. Motion reconstruction using sparse accelerometer data. *ACM Trans. Graphics*, 30(3):18:1–18:12, May 2011. 7
- [22] S.-F. C. X.-M. Wu, Z. Li. Analyzing the harmonic structure in graph-based learning. In *NIPS*, pages 3129–3137, 2013. 2
- [23] A. L. Yuille and N. M. Grzywacz. The motion coherence theory. In *Proc. ICCV*, pages 344–353, 1988. 3, 5
- [24] X. Zhou and M. Belkin. Semi-supervised learning by higher order regularization. *JMLR W&CP (Proc. AISTATS)*, pages 892–900, 2011. 1, 2, 3, 6

## 2 Appendix

# Supplementary material for “Local High-order Regularization on Data Manifolds”

We show the proof of proposition 1 in the main paper and analyze the speed of the convergence of our RKHS norm-based energy estimate to the stabilized regularization energy.

## 1 Proof of proposition 1

In Riemannian normal coordinates ( $\mathbf{x} = [x^1, \dots, x^m]^\top$ ) centered at each point  $X$ , Laplacian evaluation at  $X$  becomes the trace of the Hessian matrix given as an evaluation of the Hessian operator at  $X$ . Given this, the proof is obtained by slightly modifying the techniques used in the convergence analysis of [Audibert and Tsybakov, 2007] and [Kim et al., 2013]. We include the proof for completeness.

Given an underlying probability distribution  $P$  supported by a manifold  $M$ , our goal is to show that for each point  $X \in M$ ,  $tr[Hh(\mathbf{x})]$  converges to  $\Delta f(X)$  as the size  $u(t)$  of  $\mathcal{X}_{u(t)} = \{X_1, \dots, X_{u(t)}\} \subset M$  grows.

At each data point  $X_i \in \mathcal{X}$ , the Hessian  $Hf(X_i)$  of  $f$  is estimated by fitting  $h^i$  to  $f|_{\mathcal{N}_\epsilon(X_i)}$ , where  $\mathcal{N}_\epsilon(X_i) = \mathcal{B}(X_i, \epsilon) \cap \mathcal{X}$ ,  $\mathcal{B}(X_i, \epsilon)$  is the  $\epsilon$ -neighborhood of  $X_i$  in coordinates, i.e.,  $\mathcal{B}(X, \epsilon) := \{X' : \|\mathbf{x} - \mathbf{x}'\|_{T_X(M)} \leq \epsilon\}$ ,<sup>1</sup> and  $h|_{\mathcal{S}}$  denotes the restriction of a function  $h$  on a set  $\mathcal{S}$ : The Hessian  $Hh^i(\mathbf{x}_i)$  of  $h^i$  is used as an estimate of  $Hf(X_i)$ . Since the convergence property is homogeneous, we focus only on a single point  $X_i \in M$ . For notational convenience, we will omit the index  $i$  and furthermore, we will identify a point  $X_j \in M$  with its normal coordinate representation  $\mathbf{x}_j \in T_{X_i}(M)$  at  $X_i$  whenever the latter is defined: The normal coordinate value  $\mathbf{x}_j$  of  $X_j$  is defined when  $X_j$  is included in the *injectivity radius*  $\text{inj}(X_i)$  of  $X_i$  [Lee, 1997]. Here, we assume that (for sufficiently large  $u$ )  $\mathcal{N}_\epsilon(X_i) \in \text{inj}(X_i)$ . This is possible since in a Riemannian manifold,  $\text{inj}(X_i)$  is always positive for any  $X_i \in M$ . For the points  $X_j \notin \text{inj}(X_i)$ , the corresponding normal coordinate values are assigned with 0.

Accordingly,  $\mathcal{N}_\epsilon(X_i)$  will be represented based on its elements in coordinates  $\{\mathbf{g}_1, \dots, \mathbf{g}_k\}$ . Here, we use letter  $\mathbf{g}$  instead of  $\mathbf{x}$  to stress its indexing within  $\mathcal{N}_\epsilon(X_i)$  rather than  $\mathcal{X}$ . Note that at the normal coordinate chart centered at  $X_i$ , the coordinate value  $\mathbf{x}_i$  of  $X_i$  is zero.

The coefficients of  $h$  (Eq. 10 in the main paper):

$$h(\mathbf{x}) = f(0) + \sum_{r=1}^m [a]_r x^r + \sum_{r=1, s=r}^m [b]_{r,s} x^r x^s, \quad (1)$$

are obtained by solving a weighted least squares problem centered at  $\mathbf{x}_i = 0$ :

$$\begin{aligned} A \approx B &= \arg \min_Q \|\mathbf{K}(\mathbf{X}Q - \mathbf{f})\|^2 \\ &= (\mathbf{X}^\top \mathbf{K} \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{K} \mathbf{f}, \end{aligned} \quad (2)$$

where  $\mathbf{X}$  is the design matrix containing the first and the second-order monomials of the coordinate values (centered at  $\mathbf{x}_i = 0$ ) of data points in  $\mathcal{X}$ :

$$A = \left[ \nabla f(0)^\top, \frac{1}{2} \text{vec}[Hf(0)] \right]^\top, B = [\dots, [a]_r, \dots, [b]_{r,s}, \dots]^\top, \mathbf{f} = [f(X_1), \dots, f(X_u)]^\top, \quad (3)$$

where  $\text{vec}(M)$  extracts the upper triangular elements of a symmetric matrix  $M$  and forms a vector as a linear alignment of them, and  $\mathbf{K}$  is a diagonal weight matrix with  $[\mathbf{K}]_{j,j} = K(\mathbf{x}_j, \epsilon)$  and the kernel  $K$  is defined as:

$$K(\mathbf{x}, q) = \mathbb{1}_{\|\mathbf{x}\| < q}, \quad (4)$$

<sup>1</sup>For simplicity, we use the  $\epsilon$ -neighborhood instead of  $k$  nearest neighbors  $N_k(X_i)$ . The convergence in the latter case can easily be established by enforcing  $N_k(X_i) \subset \mathcal{B}(X_i, \epsilon)$ .



with  $\mathbb{1}_{[S]}$  is the indicator function of the set  $S$ . Note that when  $X_j \notin \text{inj}(X_i)$ ,  $\mathbf{x}_j = 0$ .

The convergence of  $Hf(0)$  is established when  $\|A - B\| \rightarrow 0$  as  $u \rightarrow \infty$  and  $\epsilon \rightarrow 0$ .

First, we decompose this deviation as:

$$\|A - B\|^2 \leq \|(\mathbf{X}^\top \mathbf{K} \mathbf{X})^{-1}\|_2 \|\mathbf{K}(\mathbf{X}A - \mathbf{f})\|^2, \quad (5)$$

where we have used  $\mathbf{K} = \mathbf{K}^2 = \mathbf{K}^\top$  and  $\|C^\top C\|_2^2 = \|C^\top\|_2^2$  for any matrix  $C$ . In Eq. 5, the first term depends only on the distribution  $P$  on  $M$  and it is upper bounded as:

$$\|(\mathbf{X}^\top \mathbf{K} \mathbf{X})^{-1}\|_2 \leq \frac{1}{\|u\epsilon^m \mathcal{E}^{-1} \bar{B} \mathcal{E}^{-1}\|_2} \leq \frac{1}{u\epsilon^{m+4} \lambda_{\bar{B}}}, \quad (6)$$

where

$$\begin{aligned} \mathcal{E} &= \text{diag}([1/\epsilon, \dots, 1/\epsilon, 1/\epsilon^2, \dots, 1/\epsilon^2]^\top), \\ \bar{B} &= \frac{1}{u\epsilon^m} \sum_{j=1}^u X(\mathbf{x}_j/\epsilon)^\top X(\mathbf{x}_j/\epsilon) K(\mathbf{x}_j, \epsilon), \\ X(\mathbf{x}) &= [x^1, \dots, x^m, \dots, x^r x^s, \dots] \in \mathbb{R}^D, \end{aligned}$$

$D = m + \frac{m(m+1)}{2}$ , and  $\lambda_{\bar{B}}$  is the smallest eigenvalue of  $\bar{B}$ . The remainder of this section quantifies  $\|\mathbf{K}(\mathbf{X}A - \mathbf{f})\|^2$  and  $\lambda_{\bar{B}}$  based on the two regularity assumptions on the Hessian operator  $H$  and the probability distribution  $P$ .

### 1.1 Quantifying $\|\mathbf{K}(\mathbf{X}A - \mathbf{f})\|^2$

The deviation between the second-order approximation  $\mathbf{K}A$  and  $\mathbf{K}f$  depends on the smoothness of  $f$ . In particular, we can quantify them based on the boundedness of  $Hf$ :

**Lemma 1 ([Belward et al., 2008])** *Suppose that the Hessian ( $Hf(\mathbf{a}) := H_f(\mathbf{a})$ ) is Lipschitz continuous with the Lipschitz constant  $\gamma$ . Then*

$$\|\mathbf{K}(\mathbf{X}A - \mathbf{f})\|_2^2 = C_1 \gamma^2 k \epsilon^6, \quad (7)$$

with a constant  $C_1 > 0$  where  $k$  is the size of  $\mathcal{N}_\epsilon(0)$ .

**Proof:** For the simplicity of exposition, let's represent each element of  $\mathcal{N}_\epsilon(0) = \{\mathbf{g}_1, \dots, \mathbf{g}_k\}$  based on its scale and the normalized coordinate values:  $\mathbf{g}_j = s_j \mathbf{v}_j$  with  $\|\mathbf{v}_j\| = 1$ .

Applying the first-order Taylor series expansion formula to  $f$  expanded at 0 gives for each point  $\mathbf{g}_i$ ,

$$\begin{aligned} f(s_j \mathbf{v}_j) - f(0) - s_j \mathbf{v}_j^\top \nabla f(0) &= \int_0^1 (1-t) s_j \mathbf{v}_j^\top H_f(s_j \mathbf{v}_j t) s_j \mathbf{v}_j dt, \\ \Leftrightarrow f(s_j \mathbf{v}_j) - \frac{1}{2} s_j \mathbf{v}_j^\top H_f(0) s_j \mathbf{v}_j - f(0) - s_j \mathbf{v}_j^\top \nabla f(0) &= \int_0^1 (1-t) s_j \mathbf{v}_j^\top (H_f(s_j \mathbf{v}_j t) - H_f(0)) s_j \mathbf{v}_j dt, \end{aligned} \quad (8)$$

where  $\nabla f := \nabla f$ .

Substituting the definition of  $A$  (Equation 3) into (8) gives  $[\mathbf{K}(\mathbf{X}A - \mathbf{f})]_j = 0$  when  $[\mathbf{K}]_{j,j} = 0$  and

$$\begin{aligned} |[\mathbf{K}(\mathbf{X}A - \mathbf{f})]_j| &= \left| \frac{1}{2} s_j \mathbf{v}_j^\top H_f(0) s_j \mathbf{v}_j - f(s_j \mathbf{v}_j) + f(0) + s_j \mathbf{v}_j^\top \nabla f(0) \right| \\ &\leq \int_0^1 |(1-t) s_j \mathbf{v}_j^\top (H_f(0) - H_f(s_j \mathbf{v}_j t)) s_j \mathbf{v}_j| dt \\ &= \frac{1}{6} \gamma s_j^3, \text{ otherwise.} \end{aligned}$$

Then

$$\|\mathbf{K}(\mathbf{X}A - \mathbf{f})\|^2 = \sum_{j=1}^u [\mathbf{K}(\mathbf{X}A - \mathbf{f})]_j^2 \leq \frac{1}{36} k \gamma^2 \epsilon^6, \quad (9)$$

where we used the fact that only  $k$  summands are non-zero and  $s_j \leq \epsilon$ .  $\square$

Substituting Eqs. 9 and 6 into Eq. 5 gives

$$\|A - B\|^2 \leq \frac{1}{36} \frac{1}{\lambda_{\bar{B}}} \frac{k \gamma^2}{u \epsilon^{m-2}}. \quad (10)$$

## 1.2 Quantifying $\lambda_{\overline{B}}$

Here, we adopt the results of [Audibert and Tsybakov, 2007] to construct a lower bound of  $\lambda_{\overline{B}}$ . Applying this result requires a certain regularity assumption on the underlying probability distribution  $P$  on  $M$  ([Audibert and Tsybakov, 2007]; Definition 1 of the main paper):

For some constants  $c_0, \epsilon_0 > 0$ , we will say that a Lebesgue measurable set  $A \subset \mathbb{R}^m$  is  $(c_0, \epsilon_0)$ -regular if

$$\lambda[A \cap \mathcal{B}(\mathbf{x}, \epsilon)] \geq c_0 \lambda[\mathcal{B}(\mathbf{x}, \epsilon)], \quad \forall \epsilon \in [0, \epsilon_0], \forall \mathbf{x} \in A, \quad (11)$$

where  $\lambda[S]$  is the Lebesgue measure of  $S \subset \mathbb{R}^m$ . We fix constants  $c_0, \epsilon_0 > 0$  and  $0 < \mu_{\min} < \mu_{\max} < \infty$  and a compact  $\mathcal{C} \subset \mathbb{R}^m$ . We say that the *strong density assumption* is satisfied if the distribution  $P$  is supported on a compact  $(c_0, \epsilon_0)$ -regular set  $A \subseteq \mathcal{C}$  and has a density  $\mu$  w.r.t. the Lebesgue measure bounded away from zero and infinity on  $A$  (between  $\mu_{\min}$  and  $\mu_{\max}$ )

$$\mu_{\min} \leq \mu(\mathbf{x}) \leq \mu_{\max}, \quad \forall \mathbf{x} \in A \text{ and } \mu(\mathbf{x}) = 0 \text{ otherwise.} \quad (12)$$

**Theorem 1 ([Audibert and Tsybakov, 2007])** *Let  $P$  satisfy the strong density assumption. Then, there exist constants  $C_2, \mu_0 > 0$  such that for any  $0 < \epsilon \leq \epsilon_0$  and any  $n \geq 1$ ,*

$$P^{\otimes n}(\lambda_{\overline{B}} \leq \mu_0) \leq 2D \exp(-C_2 n \epsilon^m), \quad (13)$$

where  $P^{\otimes n}$  is the product probability measure according to which the sample is distributed.

Combining Eq. 13 and Eq. 10, we obtain that there are positive constants  $C_1, C_2, \mu_0$  with probability larger than  $1 - (m^2 + 3m) \exp(-C_2 n \epsilon^m)$ ,

$$\|A - B\|^2 \leq \frac{C_1}{\mu_0} \frac{k\gamma^2}{u\epsilon^{m-2}}. \quad (14)$$

Adopting the strong density assumption, the probability  $P_\epsilon$  of sampling a data point from the  $\epsilon$ -neighborhood of  $\mathbf{x}_i = 0$  (which is assumed to be zero) is

$$P_\epsilon = \int_A \mu(\mathbf{x}) \mathbb{1}_{\|\mathbf{x}\| < \epsilon} d\mathbf{x} \leq \mu_{\max} \int_A \mathbb{1}_{\|\mathbf{x}\| < \epsilon} d\mathbf{x} = \mu_{\max} v_m \epsilon^m, \quad (15)$$

where  $v_m = \lambda[\mathcal{B}(0, 1)]$  and  $A$  is the support of  $P$ .

Let's define variables  $\{\mathbb{1}_\epsilon(j)\}$

$$\mathbb{1}_\epsilon(j) = \begin{cases} 1 & \text{if } \mathbf{x}_j \in \mathcal{N}_\epsilon(0) \\ 0 & \text{otherwise.} \end{cases} \quad (16)$$

Applying Hoeffding's inequality to  $\{\mathbb{1}_\epsilon(1), \dots, \mathbb{1}_\epsilon(u)\}$  yields

$$P \left( \sum_{j=1}^u \mathbb{1}_\epsilon(j) - uP_\epsilon \geq t \right) \leq \exp \left( -\frac{2t^2}{u} \right). \quad (17)$$

Substituting Eq. 15 into Eq. 17 we obtain

$$P \left( k - (\mu_{\max} v_m) u \epsilon^m \geq t \right) \leq \exp \left( -\frac{2t^2}{u} \right), \quad (18)$$

which states that  $\frac{k}{u\epsilon^d} = \mathcal{O}(1)$ .

## 2 Convergence of the RKHS-norm based energy evaluation to the stabilized energy

In Sec. 4 of the main paper, we use the RKHS-norm based local energy evaluation (Eq. 23 of the main paper)

$$\|g^i\|_D^2 := \sum_{k=1}^{\infty} c_k \int_{T_{\mathbf{x}_i}(M)} |D^k g^i|_{\mathbf{x}}|^2 d\mathbf{x} = \|g^i\|_K^2, \quad (19)$$

as an approximation of the stabilized energy (Eq. 21 of the main paper).

This is motivated by their large-scale behaviors: As  $u \rightarrow \infty$ , the kernel parameter  $\sigma$  and the diameter of  $N_k(X_i)$  shrink toward zero, and the approximation error (i.e. deviation between the two energies) converges to zero at super-linear speed.

We first note that when we use a Gaussian kernel  $K(\mathbf{x}_j, \cdot)$  ( $X_j \in N_K(X_i)$ ; instead of  $g^i$ ), the series in Eq. 19 converges absolutely. Accordingly, the contribution of all high-order (say, from  $q$  to infinite) terms in Eq. 19 is bounded by a monotonically decreasing function  $G$  of  $q$  [Yuille and Grzywacz, 1988]: Since precisely,  $\|K(\mathbf{x}_j, \cdot)\|_D^2 = 1$ ,  $G$  is explicitly given as

$$G(q) = 1 - \sum_{k=1}^{q-1} c_k \int_{T_{X_i}(M)} |D^k g^i|_{\mathbf{x}}|^2 d\mathbf{x}. \quad (20)$$

Furthermore, the corresponding sum of the high-order terms in the stabilized energy is bounded by  $[0, G(q)]$ . Accordingly, we can determine a  $q^*$  so that the finite sum up to the  $q^*$ -th summand in Eq. 19 approximates the local energy with any given approximation accuracy level.

Secondly, for any finite order  $q' \leq q^*$ , the corresponding derivatives of a Gaussian function are given as the  $q'$ -th order Hermite polynomials multiplied by the Gaussian. Since a Gaussian function *suppresses* any polynomials, these derivatives decrease rapidly as the corresponding points of evaluation deviates from its center  $X_i$  [Kara, 2009]. The speed of this decay is controlled by the width  $\sigma^2$  of  $K$ . Accordingly, for given upper bound  $s$  on the approximation error and the integral domain  $\mathcal{U}'(X_j)$ ,  $\sigma^2$  can be determined such that the deviation between the local energy of  $K(\mathbf{x}_j, \cdot)$  (defined based on the integrals over the entire  $T_{X_i}(M)$  in Eq. 19) and its restriction to  $\mathcal{U}'(X_j)$  becomes smaller than  $s$ . This can be shown by straightforwardly evaluating integrals: For instance when  $m = 1$  and  $(-\mu, \mu) = \mathcal{U}'(0)$ , ( $X_j$  is assumed to be 0 without loss of generality), the integral of the first-order norm of  $K(x) = \exp(-\frac{x^2}{\sigma^2})$  over the entire domain  $\mathbb{R} \sim T_0(M)$  is<sup>2</sup>

$$\int_{-\infty}^{\infty} \left| \frac{\partial}{\partial x} \exp\left(-\frac{x^2}{\sigma^2}\right) \right|^2 dx = \int_{-\infty}^{\infty} \frac{4x^2}{\sigma^4} \exp\left(-2\frac{x^2}{\sigma^2}\right) dx = \sqrt{\frac{\pi}{2}} \frac{1}{\sigma}. \quad (21)$$

and the corresponding integral restricted to  $\mathcal{U}'(0)$  is obtained as

$$\begin{aligned} \int_{-u}^u \left| \frac{\partial}{\partial x} \exp\left(-\frac{x^2}{\sigma^2}\right) \right|^2 dx &= \sqrt{\frac{\pi}{2}} \frac{1}{\sigma} \operatorname{erf}\left(\sqrt{2}\frac{\mu}{\sigma}\right) - \frac{\mu\sqrt{\pi}}{\sigma^2} \operatorname{erf}'\left(\sqrt{2}\frac{u}{\sigma}\right) \\ &= \sqrt{\frac{\pi}{2}} \frac{1}{\sigma} \operatorname{erf}\left(\sqrt{2}\frac{\mu}{\sigma}\right) - \frac{2\mu}{\sigma^2} \exp\left(-2\frac{u^2}{\sigma^2}\right), \end{aligned} \quad (22)$$

where *erf* is the error function of the standard Gaussian distribution. We used the *differentiation under the integral sign* (w.r.t.  $\sigma$ ) technique to calculate the integrals in Eqs. 21 and 22. The approximation error of the first-order term is then obtained as,

$$\begin{aligned} c_1 &\left( \int_{-\infty}^{\infty} \left| D^1 \exp\left(-\frac{x^2}{\sigma^2}\right) \right|^2 dx - \int_{-u}^u \left| D^1 \exp\left(-\frac{x^2}{\sigma^2}\right) \right|^2 dx \right) \\ &= \frac{\sigma}{2} \sqrt{\frac{\pi}{2}} \left[ 1 - \operatorname{erf}\left(\sqrt{2}\frac{\mu}{\sigma}\right) - 2\sqrt{\frac{2}{\pi}} \frac{\mu}{\sigma} \exp\left(-2\frac{u^2}{\sigma^2}\right) \right]. \end{aligned} \quad (23)$$

Similarly, the approximation error of the second-order term is given as,

$$\begin{aligned} c_2 &\left( \int_{-\infty}^{\infty} \left| D^2 \exp\left(-\frac{x^2}{\sigma^2}\right) \right|^2 dx - \int_{-u}^u \left| D^2 \exp\left(-\frac{x^2}{\sigma^2}\right) \right|^2 dx \right) \\ &= 3\sigma \sqrt{\frac{\pi}{2}} \left[ 1 - \operatorname{erf}\left(\sqrt{2}\frac{u}{\sigma}\right) + \frac{1}{3} \sqrt{\frac{2}{\pi}} \exp\left(-2\frac{u^2}{\sigma^2}\right) \left( \left(\frac{2u}{\sigma}\right)^3 - \frac{2u}{\sigma} \right) \right]. \end{aligned} \quad (24)$$

<sup>2</sup>Since the Gaussian RKHS energy is shift invariant, we assume that  $X_i = 0$  without loss of generality.

For both the first- and the second-order errors, the normalized errors (terms in the outer parentheses in Eqs. 23 and 24) are bounded by  $[0, 1]$ . When  $\sigma$  is large, the exponential terms in the normalized errors dominates and accordingly, the normalized errors tend to be 1. In this *worse* case, the approximation error decreases linearly with respect to  $\sigma$  decrease. However, when  $\sigma$  is already small, *erf* dominates in the normalized errors<sup>3</sup> and therefore, in this case, the corresponding approximation errors decrease super-linearly. Since iteratively taking the derivatives of a Gaussian with respect to  $\sigma^2$  yields polynomials, repeatedly applying the differentiation under the integral sign technique shows that all higher-order terms behave exactly the same way: for each  $k$ , the approximation error is given as a linear term  $\alpha^k \sigma$  (with  $\alpha^k$  being a constant) multiplied by the normalized error which is dominated by an error function for small  $\sigma$ . Accordingly, for any order  $k$ , the approximation error decreases super-linearly.

Since  $D^k g^i$  is a kernel expansion of  $N_k(X_i)$ , its *effective* support can be limited within a neighborhood that encompass  $\{U'(X_k), \forall X_k \in N_k(X_i)\}$ . Then, we can control both  $\sigma^2$  and the diameter of  $N_k(X_i)$  so that the resulting local energy  $\|g^i\|_D^2$  well-approximates the integrand in the regularizer (Eq. 7 of the main paper).

$$\|f\|_D^2 := \int_M \sum_{k=1}^{\infty} c_k |D^k f|_X|^2 dV(X), \quad (25)$$

where

$$D^{2k} f = \Delta^k f, D^{2k+1} f = \nabla(\Delta^k f), \quad (26)$$

and where  $c_k \geq 0$ ,  $|D^{2k} f|^2 := (D^{2k} f)^2$ , and  $|D^{2k+1} f|^2 := g(D^{2k+1} f, D^{2k+1} f)$ .

## References

- [Audibert and Tsybakov, 2007] Audibert, J.-Y. and Tsybakov, A. B. (2007). Fast learning rates for plug-in classifiers. *The Annals of Statistics*, 35(2):608–633.
- [Belward et al., 2008] Belward, J. A., Turner, I. W., and Oqielat, M. N. (2008). Numerical investigations of linear least squares methods for derivative estimation. *ANZIAM Journal*, 50:C844–C857.
- [Kara, 2009] Kara, M. (2009). An analytical expression for arbitrary derivatives of Gaussian functions  $\exp(ax^2)$ . *Internal Journal of Physical Sciences*, 4(4):247–249.
- [Kim et al., 2013] Kim, K. I., Tompkin, J., and Theobalt, C. (2013). Curvature-aware regularization on Riemannian submanifolds. In *Proc. ICCV*, pages 881–888.
- [Lee, 1997] Lee, J. M. (1997). *Riemannian Manifolds- An Introduction to Curvature*. Springer, New York.
- [Yuille and Grzywacz, 1988] Yuille, A. L. and Grzywacz, N. M. (1988). The motion coherence theory. In *Proc. ICCV*, pages 344–353.

---

<sup>3</sup>Note, the exponential term dominates every polynomials in the third summand.

Below you find a list of the most recent research reports of the Max-Planck-Institut für Informatik. Most of them are accessible via WWW using the URL <http://www.mpi-inf.mpg.de/reports>. Paper copies (which are not necessarily free of charge) can be ordered either by regular mail or by e-mail at the address below.

Max-Planck-Institut für Informatik  
 – Library and Publications –  
 Campus E 1 4

D-66123 Saarbrücken

E-mail: [library@mpi-inf.mpg.de](mailto:library@mpi-inf.mpg.de)

---

MPI-I-2013-RG1-002	P. Baumgartner, U. Waldmann	Hierarchic superposition with weak abstraction
MPI-I-2013-5-002	F. Makari, B. Awerbuch, R. Gemulla, R. Khandekar, J. Mestre, M. Sozio	A distributed algorithm for large-scale generalized matching
MPI-I-2013-1-001	C. Huang, S. Ott	New results for non-preemptive speed scaling
MPI-I-2012-RG1-002	A. Fietzke, E. Kruglov, C. Weidenbach	Automatic generation of inductive invariants by SUP(LA)
MPI-I-2012-RG1-001	M. Suda, C. Weidenbach	Labelled superposition for PLTL
MPI-I-2012-5-004	F. Alvanaki, S. Michel, A. Stupar	Building and maintaining halls of fame over a database
MPI-I-2012-5-003	K. Berberich, S. Bedathur	Computing n-gram statistics in MapReduce
MPI-I-2012-5-002	M. Dylla, I. Miliaraki, M. Theobald	Top-k query processing in probabilistic databases with non-materialized views
MPI-I-2012-5-001	P. Miettinen, J. Vreeken	MDL4BMF: Minimum Description Length for Boolean Matrix Factorization
MPI-I-2012-4-001	J. Kerber, M. Bokeloh, M. Wand, H. Seidel	Symmetry detection in large scale city scans
MPI-I-2011-RG1-002	T. Lu, S. Merz, C. Weidenbach	Towards verification of the pastry protocol using TLA+
MPI-I-2011-5-002	B. Taneva, M. Kacimi, G. Weikum	Finding images of rare and ambiguous entities
MPI-I-2011-5-001	A. Anand, S. Bedathur, K. Berberich, R. Schenkel	Temporal index sharding for space-time efficiency in archive search
MPI-I-2011-4-005	A. Berner, O. Burghard, M. Wand, N.J. Mitra, R. Klein, H. Seidel	A morphable part model for shape manipulation
MPI-I-2011-4-002	K.I. Kim, Y. Kwon, J.H. Kim, C. Theobald	Efficient learning-based image enhancement : application to compression artifact removal and super-resolution
MPI-I-2011-4-001	M. Granados, J. Tompkin, K. In Kim, O. Grau, J. Kautz, C. Theobald	How not to be seen inpainting dynamic objects in crowded scenes
MPI-I-2010-RG1-001	M. Suda, C. Weidenbach, P. Wischniewski	On the saturation of YAGO
MPI-I-2010-5-008	S. Elbassuoni, M. Ramanath, G. Weikum	Query relaxation for entity-relationship search
MPI-I-2010-5-007	J. Hoffart, F.M. Suchanek, K. Berberich, G. Weikum	YAGO2: a spatially and temporally enhanced knowledge base from Wikipedia
MPI-I-2010-5-006	A. Broschart, R. Schenkel	Real-time text queries with tunable term pair indexes
MPI-I-2010-5-005	S. Seufert, S. Bedathur, J. Mestre, G. Weikum	Bonsai: Growing Interesting Small Trees
MPI-I-2010-5-004	N. Preda, F. Suchanek, W. Yuan, G. Weikum	Query evaluation with asymmetric web services
MPI-I-2010-5-003	A. Anand, S. Bedathur, K. Berberich, R. Schenkel	Efficient temporal keyword queries over versioned text
MPI-I-2010-5-002	M. Theobald, M. Sozio, F. Suchanek, N. Nakashole	URDF: Efficient Reasoning in Uncertain RDF Knowledge Bases with Soft and Hard Rules
MPI-I-2010-5-001	K. Berberich, S. Bedathur, O. Alonso, G. Weikum	A language modeling approach for temporal information needs

MPI-I-2010-1-001	C. Huang, T. Kavitha	Maximum cardinality popular matchings in strict two-sided preference lists
MPI-I-2009-RG1-005	M. Horbach, C. Weidenbach	Superposition for fixed domains
MPI-I-2009-RG1-004	M. Horbach, C. Weidenbach	Decidability results for saturation-based model building
MPI-I-2009-RG1-002	P. Wischniewski, C. Weidenbach	Contextual rewriting
MPI-I-2009-RG1-001	M. Horbach, C. Weidenbach	Deciding the inductive validity of $\forall\exists^*$ queries
MPI-I-2009-5-007	G. Kasneci, G. Weikum, S. Elbassuoni	MING: Mining Informative Entity-Relationship Subgraphs
MPI-I-2009-5-006	S. Bedathur, K. Berberich, J. Dittrich, N. Mamoulis, G. Weikum	Scalable phrase mining for ad-hoc text analytics
MPI-I-2009-5-005	G. de Melo, G. Weikum	Towards a Universal Wordnet by learning from combined evidenc
MPI-I-2009-5-004	N. Preda, F.M. Suchanek, G. Kasneci, T. Neumann, G. Weikum	Coupling knowledge bases and web services for active knowledge
MPI-I-2009-5-003	T. Neumann, G. Weikum	The RDF-3X engine for scalable management of RDF data
MPI-I-2009-5-003	T. Neumann, G. Weikum	The RDF-3X engine for scalable management of RDF data
MPI-I-2009-5-002	M. Ramanath, K.S. Kumar, G. Ifrim	Generating concise and readable summaries of XML documents
MPI-I-2009-4-006	C. Stoll	Optical reconstruction of detailed animatable human body models
MPI-I-2009-4-005	A. Berner, M. Bokeloh, M. Wand, A. Schilling, H. Seidel	Generalized intrinsic symmetry detection
MPI-I-2009-4-004	V. Havran, J. Zajac, J. Drahokoupil, H. Seidel	MPI Informatics building model as data for your research
MPI-I-2009-4-003	M. Fuchs, T. Chen, O. Wang, R. Raskar, H.P.A. Lensch, H. Seidel	A shaped temporal filter camera
MPI-I-2009-4-002	A. Tevs, M. Wand, I. Ihrke, H. Seidel	A Bayesian approach to manifold topology reconstruction
MPI-I-2009-4-001	M.B. Hullin, B. Ajdin, J. Hanika, H. Seidel, J. Kautz, H.P.A. Lensch	Acquisition and analysis of bispectral bidirectional reflectance distribution functions
MPI-I-2008-RG1-001	A. Fietzke, C. Weidenbach	Labelled splitting
MPI-I-2008-5-004	F. Suchanek, M. Sozio, G. Weikum	SOFI: a self-organizing framework for information extraction
MPI-I-2008-5-003	G. de Melo, F.M. Suchanek, A. Pease	Integrating Yago into the suggested upper merged ontology
MPI-I-2008-5-002	T. Neumann, G. Moerkotte	Single phase construction of optimal DAG-structured QEPs
MPI-I-2008-5-001	G. Kasneci, M. Ramanath, M. Sozio, F.M. Suchanek, G. Weikum	STAR: Steiner tree approximation in relationship-graphs
MPI-I-2008-4-003	T. Schultz, H. Theisel, H. Seidel	Crease surfaces: from theory to extraction and application to diffusion tensor MRI
MPI-I-2008-4-002	D. Wang, A. Belyaev, W. Saleem, H. Seidel	Estimating complexity of 3D shapes using view similarity
MPI-I-2008-1-001	D. Ajwani, I. Malinger, U. Meyer, S. Toledo	Characterizing the performance of Flash memory storage devices and its impact on algorithm design
MPI-I-2007-RG1-002	T. Hillenbrand, C. Weidenbach	Superposition for finite domains
MPI-I-2007-5-003	F.M. Suchanek, G. Kasneci, G. Weikum	Yago : a large ontology from Wikipedia and WordNet
MPI-I-2007-5-002	K. Berberich, S. Bedathur, T. Neumann, G. Weikum	A time machine for text search
MPI-I-2007-5-001	G. Kasneci, F.M. Suchanek, G. Ifrim, M. Ramanath, G. Weikum	NAGA: searching and ranking knowledge