Incomplete Gamma Kernels: Generalizing Locally Optimal Projection Operators

Patrick Stotko, Michael Weinmann, and Reinhard Klein

Abstract—We present incomplete gamma kernels, a generalization of Locally Optimal Projection (LOP) operators. In particular, we reveal the relation of the classical localized L_1 estimator, used in the LOP operator for point cloud denoising, to the common Mean Shift framework via a novel kernel. Furthermore, we generalize this result to a whole family of kernels that are built upon the incomplete gamma function and each represents a localized L_p estimator. By deriving various properties of the kernel family concerning distributional, Mean Shift induced, and other aspects such as strict positive definiteness, we obtain a deeper understanding of the operator's projection behavior. From these theoretical insights, we illustrate several applications ranging from an improved Weighted LOP (WLOP) density weighting scheme and a more accurate Continuous LOP (CLOP) kernel approximation to the definition of a novel set of robust loss functions. These incomplete gamma losses include the Gaussian and LOP loss as special cases and can be applied to various tasks including normal filtering. Furthermore, we show that the novel kernels can be included as priors into neural networks. We demonstrate the effects of each application in a range of quantitative and qualitative experiments that highlight the benefits induced by our modifications.

Index Terms—Kernels, Locally Optimal Projection, Mean Shift, point clouds, point cloud denoising, projection operators, robust loss functions, surface reconstruction, theory

1 Introduction

Digital 3D scene models have become a crucial prerequisite for numerous applications in entertainment, advertisement, design, architecture, autonomous systems, and cultural heritage. In this context, the accurate digitization of real-world objects and scenes is of great relevance and offers new opportunities regarding a variety of tasks including AR/VR-based inspection and collecting realistic training data for tasks in robotics, autonomous driving, aerial or satellite surveys. Aside from professional scanning campaigns with expensive laser scanning equipment, there has also been an increasing trend towards more practical scene capture with consumer-grade hardware such as passive purely image-based scene scanning using Structure-from-Motion and Multi-view Stereo approaches, or with respective cheaper active time-of-flight depth sensors that have meanwhile even been integrated into numerous mobile devices. However, the use of passive scene scanning or active scanning based on cheap hardware with low sensor quality and low sensor resolution induces noise in the capture process and thereby results in noisy point clouds and a low number of points that might not preserve finer geometric details, which, in turn, may lead to artifacts in the registration and subsequent surface reconstruction procedures. Furthermore, the limited accessibility of capture conditions as well as occlusions induce holes and highly irregular samplings and distributions of the captured data. These challenges result in an increasing interest in robust filtering techniques that are capable of handling noise, outliers, registration artifacts, as well as irregularly-sampled and missing data and can provide a clean, denoised, and uniformly resampled point cloud suitable for high-fidelity surface reconstruction.

Among others, the Locally Optimal Projection (LOP) operator [1] has gained a lot of attention in recent years due to its benefit of not relying on a well-defined surface parametrization or a piecewise planar approximation and, meanwhile, there has been a whole series of further extensions of this approach [2], [3], [4], [5]. Furthermore, many learning-based approaches also aim at projecting the noisy data onto a (latent) denoised manifold [6], [7]. Therefore, investigations towards the unification of traditional approaches with their respective regularization techniques might be of great relevance for future learning-based approaches as well. Even further, traditional techniques, and particularly those approaches that incorporate probabilistic modeling of the data such as the density-based Mean Shift clustering method [8], [9], [10], become more and more relevant in modern deep learning methods. Besides their application to structure the latent space representation of the data within encoder-decoder approaches [11], there is even a direct relation between the Mean Shift approach and denoising autoencoders [12]. In particular, as the output of an optimal denoising autoencoder corresponds to the local mean of the true data density [13], the autoencoder loss can be interpreted as a Mean Shift vector [12]. However, to the best of our knowledge, this observation has not yet been explored in the context of point cloud denoising. Hence, relating traditional concepts to modern deep learning methods might not only lead to a more explainable behavior of the latter but also allow increasing the resulting performance. In turn, this relies on the better understanding of the

- *P. Stotko and R. Klein are with the Institute of Computer Science II – Visual Computing, University of Bonn, Germany. E-mail: stotko@cs.uni-bonn.de, rk@cs.uni-bonn.de*
- *M. Weinmann is with the Department of Intelligent Systems, Delft University of Technology, The Netherlands E-mail: m.weinmann@tudelft.nl*

Manuscript received MM DD, 20YY; revised MM DD, 20YY.

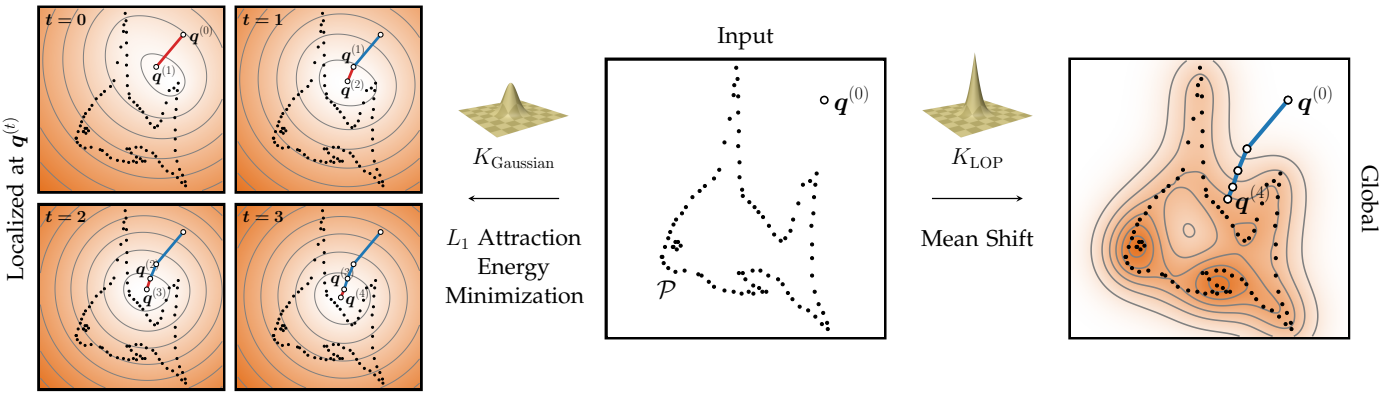


Fig. 1. Relation between LOP and Mean Shift in the example of the *2D Fish* model. Minimizing the *localized* L_1 attraction energy with the Gaussian kernel K_{Gaussian} (*left*) results in the same trajectory $\boldsymbol{q}^{(t)}$ as applying Mean Shift on a *global* kernel density estimate with the kernel K_{LOP} (*right*).

relationship between previous (traditional) techniques.

In this paper, we investigate the theoretical relationship of projection-based point cloud denoising approaches with their respective properties and show that these are unified within the common probabilistic Mean Shift framework. In particular, the key contributions of our work are:

- We reveal the relation of the classical localized L_1 estimator used in LOP to the Mean Shift framework via a novel kernel K_{LOP} and introduce the family of *incomplete gamma kernels* K_Γ as a generalization of this result where each kernel represents a localized L_p estimator (see Section 3).
- We derive various properties of the kernel family concerning distributional, Mean Shift induced, and other aspects such as strict positive definiteness to obtain a deeper understanding of the operator's projection behavior (see Section 4).
- We demonstrate that leveraging the derived theoretical insights enables several applications including an improved Weighted LOP (WLOP) density weighting scheme, a more accurate Continuous LOP (CLOP) kernel approximation, the derivation of *incomplete gamma losses*, a set of novel robust loss functions, as well as neural network priors (see Section 5).

In our evaluation, we demonstrate the benefits induced by our modifications in a range of quantitative and qualitative experiments. Furthermore, the theoretical insights of our investigations with their proven effect may be of great relevance also for future learning-based approaches. The source code is available at https://github.com/stotko/incomplete-gamma-kernels.

2 Related Work

In the following, we provide a review of geometric and learning-based denoising approaches. Furthermore, we also review seminal work regarding the theory and application of the Mean Shift framework due to its relationship to LOP approaches that we will demonstrate later.

2.1 Geometric Denoising Approaches

Following early approaches such as the local fitting of tangent planes [14] or using radial basis functions [15], respective developments particularly focused on projection-based methods, sparsity-based methods and non-local methods.

Projection-based Methods. These approaches rely on the assumption of an underlying smooth surface and the projection of noisy data points onto the estimated local surface. For this purpose, respective approaches apply moving least squares (MLS) based methods [16], [17], [18], [19], robust principal component analysis (RPCA) [20] and moving robust principal component analysis (MRPCA) [21], or locally optimal projection based operators where the LOP operator [1] has been extended in terms of Weighted LOP (WLOP) [2], Feature LOP (FLOP) [3], Continuous LOP (CLOP) [5], Edge-Aware Resampling (EAR) [4] and a Gaussian mixture model inspired projection operator [22]. The latter has been demonstrated to be capable of resampling point clouds while preserving features due to the additional guidance of filtered normals.

Sparsity-based Methods. This class of approaches relies on the assumption that objects can be represented in terms of piecewise smooth surfaces with sparse features. Respective denoising techniques include L_0-norm [23], [24] and L_1-norm minimization [21], [25], [26], sparse dictionary learning [27] as well as patch-based or feature-based graph Laplacian regularization [28], [29], [30], graph-based point cloud denoising based on jointly leveraging geometry and color information [31], guided filtering based on normal information followed by a L_1-medial skeleton extraction to get the sharp structure of the surface [32] as well as leveraging gravitational feature functions [33]. In the context of denoising dynamic point clouds, Hu *et al.* [34] explored the temporal coherence of spatio-temporal graphs with respect to the underlying surface, where a respective manifold-to-manifold distance has been introduced. Furthermore, data-driven exemplar priors have been used for surface reconstruction [35], where the sparsity of local shapes from a collection of 3D objects has been explored.

Non-local Methods. In contrast to the previous classes, these approaches rely on the assumption that geometric statistics are (approximately) shared by certain surface patches of a 3D model, i.e. local surface denoising is conducted based on collected neighborhoods with similar geometry [36], [37], [38], [39]. However, the definition of a suitable metric as well as the regular representation of local surface structures remain challenging. Furthermore, density-

based point cloud denoising has been approached by first applying particle-swarm based optimization for kernel density estimation followed by a Mean Shift clustering-based outlier removal and a final bilateral mesh filtering [40].

2.2 Learning-based Denoising Approaches

Recent works more and more leverage deep learning for point cloud denoising as well as surface reconstruction from point clouds. Examples include approaches for point cloud consolidation and resampling such as PointNet [41], PointNet++ [42], patch-based progressive point cloud upsampling [43] as well as the unification of the considerations of densifying, denoising and completing point clouds [44]. Other approaches followed the principles of initially projecting the points onto coarse-level local reference planes and applying a subsequent refinement [45] or the initial removal of outliers before conducting the denoising [46]. Further approaches include edge-aware point cloud consolidation [47], adversarial defense [48], graph-convolutional methods [49], unsupervised approaches such as Total Denoising [50], gradient field based denoising [51], [52], [53], differentiable approaches [54], [55], [56] as well as manifold learning based on encoder-decoder architectures [6], [7]. Non-local self-similarities have also been considered to define neural self-priors that capture geometric repetitions [57], capture semantically related non-local features [58], or apply self-correction by allowing the model to capture structural and contextual information from initially disorganized parts [59]. Furthermore, normalizing flows have been applied to the learn the distribution of noisy points and disentangle noise from the latent space [60]. In addition, the feature-aware recurrent point cloud denoising network (RePCD-Net) [61] combines a recurrent network architecture for noise removal with multi-scale feature aggregation and propagation and a feature-aware Chamfer distance loss.

2.3 Mean Shift Approaches

The Mean Shift approach [8] is a well-studied local mode-seeking method with diverse applications including data clustering [9], [10], [62], [63], image filtering [10], segmentation [10], [64], denoising [12], [65], and object tracking [64]. Tremendous effort has been spent to study its convergence behavior [9], [10], [66], [67], [68], [69] which culminated in a rigorous set of properties proven by Yamasaki and Tanaka [70]. Recently, Mean Shift clustering has also been applied in the latent space of neural encoder-decoder approaches to achieve a better structured data representation [11]. Furthermore, the connection between the Mean Shift approach and denoising autoencoders [71] has been revealed by Bigdeli *et al.* [12], who leveraged the observation that the output of an optimal denoising autoencoder (DAE) is a local mean of the true data density [13] to show that that the autoencoder loss is a Mean Shift vector and to use the respective magnitude to define a prior for image restoration.

3 Background

Before deriving our proposed kernel family as a generalization of LOP in the context of Mean Shift, we first provide a brief introduction into the concepts of both approaches.

3.1 Mean Shift

The basic objective of Mean Shift [8] is to find the modes of a probabilistic distribution f which has been observed by a (sparse) set of points $\mathcal{P} = \{\boldsymbol{p}_i \in \mathbb{R}^d\}$ in a d-dimensional vector space. Due to this abstract formulation, it has been applied to various computer vision problems by choosing an appropriate application-specific feature space, e.g. the $L^*u^*v^*$ color space for image filtering or segmentation [10]. In order to analyze the unknown distribution f at any point $\boldsymbol{q} \in \mathbb{R}^d$, it is modeled by a kernel density estimate:

$$\hat{f}_{\mathcal{P},K}(\boldsymbol{q}) = \frac{1}{|\mathcal{P}|\, h^d} \sum_i K\left(\tfrac{\boldsymbol{p}_i - \boldsymbol{q}}{h}\right) \tag{1}$$

Here, h denotes the kernel window size and K a kernel that is non-negative ($K(\boldsymbol{x}) \geq 0$), normalized ($\int_{\mathbb{R}^d} K(\boldsymbol{x})\,\mathrm{d}\boldsymbol{x} = 1$), and radially symmetric ($K(\boldsymbol{x}) = c_K\, k(\|\boldsymbol{x}\|^2)$). The 1-dimensional function k defined in the symmetry constraint, where c_K denotes a normalization constant, is called the kernel profile of K and plays an important role in the analysis of Mean Shift [70]. Furthermore, the gradient of the kernel density estimate

$$\nabla \hat{f}_{\mathcal{P},K}(\boldsymbol{q}) = \frac{2}{|\mathcal{P}|\, h^{d+2}} \frac{c_K}{c_G} \sum_i G\left(\tfrac{\boldsymbol{p}_i - \boldsymbol{q}}{h}\right) (\boldsymbol{p}_i - \boldsymbol{q}) \tag{2}$$

can be derived using the kernel $G(\boldsymbol{x}) = c_G\, g(\|\boldsymbol{x}\|^2)$ with the normalization constant c_G and its corresponding profile $g(x) = -\frac{\mathrm{d}}{\mathrm{d}x} k(x)$. Based on these two functions, the main component of the algorithm for finding the modes is the Mean Shift vector [10]

$$\boldsymbol{m}_{\mathcal{P},G}(\boldsymbol{q}) = \frac{h^2}{2} \frac{c_G}{c_K} \frac{\nabla \hat{f}_{\mathcal{P},K}(\boldsymbol{q})}{\hat{f}_{\mathcal{P},G}(\boldsymbol{q})} = \frac{\sum_i G\left(\tfrac{\boldsymbol{p}_i - \boldsymbol{q}}{h}\right) (\boldsymbol{p}_i - \boldsymbol{q})}{\sum_i G\left(\tfrac{\boldsymbol{p}_i - \boldsymbol{q}}{h}\right)} \tag{3}$$

which describes the gradient vector normalized with respect to the kernel G. In particular, it directly determines the updated point at each time step t by the corresponding fixed-point iteration $\boldsymbol{q}^{(t+1)} = \boldsymbol{q}^{(t)} + \boldsymbol{m}_{\mathcal{P},G}(\boldsymbol{q}^{(t)})$ which performs gradient ascent on the kernel density estimate $\hat{f}_{\mathcal{P},K}$.

3.2 Locally Optimal Projection

Unlike approaches that are based on the probabilistic concept of Mean Shift to tackle various different computer vision tasks, the problem of denoising point clouds has also been of great relevance for the computer graphics community. Many advanced solutions have been independently developed there, especially methods leveraging the robust L_1 median. To efficiently compute the unique global solution in these median-based formulations, the iterative Weiszfeld algorithm [72] has become a popular and commonly used choice. Specifically, it also marks the foundation of LOP [1] as a robust localized 3D projection operator. Given a set of noisy 3D target points $\mathcal{P} = \{\boldsymbol{p}_i \in \mathbb{R}^3\}$ sampled from a smooth surface $\mathcal{S}$, the task here consists in projecting and uniformly distributing an independent set of 3D points $\mathcal{Q} = \{\boldsymbol{q}_j \in \mathbb{R}^3\}$ onto the unknown surface $\mathcal{S}$ which is defined by the observations $\mathcal{P}$ only. This can be expressed in terms of an energy formulation

$$E(\mathcal{Q}) = \sum_j E_{\mathrm{LOP}}(\boldsymbol{q}_j) + E_{\mathrm{rep}}(\boldsymbol{q}_j) \tag{4}$$

based on an attraction and a repulsion term

$$E_{\mathrm{LOP}}(\boldsymbol{q}_j) = \sum_i \theta(\|\boldsymbol{p}_i - \boldsymbol{q}_j^{(t)}\|)\,\|\boldsymbol{p}_i - \boldsymbol{q}_j\| \tag{5}$$

$$E_{\mathrm{rep}}(\boldsymbol{q}_j) = \lambda_j \sum_{i,i\neq j} \theta(\|\boldsymbol{q}_i^{(t)} - \boldsymbol{q}_j^{(t)}\|)\,\eta(\|\boldsymbol{q}_i^{(t)} - \boldsymbol{q}_j\|) \tag{6}$$

where $\theta(x) = \mathrm{e}^{-x^2/(h/4)^2}$ denotes a compact localization kernel and η a decreasing regularization function penalizing small distances between projection points to ensure a uniform distribution of the projected points. Common choices of η include the originally proposed function $\eta_{\mathrm{LOP}}(x) = 1/(3x^3)$ [1] as well as the less rapidly decreasing function $\eta_{\mathrm{WLOP}}(x) = -x$ [2]. Both energy terms are balanced by weights λ_j which are chosen such that they only depend on a single, global parameter $\mu \in [0, 1/2)$. Based on the Weiszfeld algorithm, the solution to this optimization problem can be obtained by the fixed-point iteration

$$\boldsymbol{q}_j^{(t+1)} = \frac{\sum_i \alpha(\|\boldsymbol{p}_i - \boldsymbol{q}_j^{(t)}\|)\,\boldsymbol{p}_i}{\sum_i \alpha(\|\boldsymbol{p}_i - \boldsymbol{q}_j^{(t)}\|)} + \mu\,\frac{\sum_{i,i\neq j} \beta(\|\boldsymbol{q}_i^{(t)} - \boldsymbol{q}_j^{(t)}\|)\,(\boldsymbol{q}_j^{(t)} - \boldsymbol{q}_i^{(t)})}{\sum_{i,i\neq j} \beta(\|\boldsymbol{q}_i^{(t)} - \boldsymbol{q}_j^{(t)}\|)} \tag{7}$$

with kernels $\alpha(x) = \theta(x)/x$ and $\beta(x) = \theta(x)/x\,\left|\frac{\mathrm{d}}{\mathrm{d}x}\eta(x)\right|$.

3.3 Generalization via Incomplete Gamma Kernels

Although Mean Shift and Locally Optimal Projection have been separately developed from different contexts and mathematical concepts with the aim of solving a distinct problem, we can link both approaches by rewriting the update step in (7):

$$\boldsymbol{q}_j^{(t+1)} = \boldsymbol{q}_j^{(t)} + \boldsymbol{m}_{\mathcal{P},G_{\mathrm{LOP}}}(\boldsymbol{q}_j^{(t)}) - \mu\,\boldsymbol{m}_{\mathcal{Q}_j^{(t)},G_{\mathrm{rep}}}(\boldsymbol{q}_j^{(t)}) \tag{8}$$

This reveals that the LOP operator is a combination of two Mean Shift steps in the 3D space: 1) a standard Mean Shift with respect to the target set $\mathcal{P}$ and G_{LOP} being the normalized kernel α; and 2) a reverse applied Blurring Mean Shift [9] where the Mean Shift vector is instead subtracted and computed from the shifted and, in turn, blurred source set $\mathcal{Q}_j^{(t)} = \mathcal{Q}^{(t)} \setminus \{\boldsymbol{q}_j^{(t)}\}$ as well as from the kernel G_{rep} corresponding to the normalized kernel β. Therefore, we can interpret the localized L_1 attraction energy minimization with a Gaussian kernel in (5) as a maximization of a global kernel density estimate (1) with respect to a different kernel K_{LOP}. An example of this relation is shown in Fig. 1.

To derive K_{LOP} as part of a novel kernel family K_Γ for the general case of the $\mathbb{R}^d$ space, we consider the 1-dimensional profile of the involved kernel G_{LOP}, i.e. α in (7), which resembles a gamma distribution $f_\Gamma(x\,|\,a,b) \propto x^{a-1}\,\mathrm{e}^{-x/b}$ with support $x \in (0,\infty)$ and parameters $a > 0, b > 0$. The profile of the actual kernel then resembles the distribution $\bar{F}_\Gamma(x\,|\,a,b) \propto \Gamma(a, x/b)$ which is the complementary CDF of f_Γ and based on the *upper incomplete gamma function* $\Gamma(a,x) = \int_x^\infty t^{a-1}\,\mathrm{e}^{-t}\,\mathrm{d}t$. Therefore, the d-dimensional kernel has the general form $K_\Gamma(\boldsymbol{x}\,|\,a,b) = c_{K_\Gamma}\,\Gamma(a, \|\boldsymbol{x}\|^2/b)$.

Since we also need to compute the respective normalization constant c_{K_Γ}, we switch the integration domain to spherical coordinates and substitute $s = r^2/b$:

$$\begin{aligned}\frac{1}{c_{K_\Gamma}} &= \int_{\mathbb{R}^d} \Gamma(a, \tfrac{\|\boldsymbol{x}\|^2}{b})\,\mathrm{d}\boldsymbol{x} = \int_\Omega \int_0^\infty \Gamma(a, \tfrac{r^2}{b})\,r^{d-1}\,\mathrm{d}r\,\mathrm{d}\boldsymbol{\Omega} \\ &= \frac{b^{\frac{d}{2}}}{2}\left[\int_\Omega \mathrm{d}\boldsymbol{\Omega}\right]\left[\int_0^\infty \Gamma(a,s)\,s^{\frac{d}{2}-1}\,\mathrm{d}s\right]\end{aligned} \tag{9}$$

Due to radial symmetry, both integrals can be solved independently. The former one describes the surface area of the d-dimensional unit sphere Ω and has the closed form $\int_\Omega \mathrm{d}\boldsymbol{\Omega} = 2\pi^{d/2}/\Gamma(d/2)$. Using the relation $\int_0^\infty \Gamma(a,x)\,x^{b-1}\,\mathrm{d}x = \Gamma(a+b)/b$ [73], we get an expression for the latter one in terms of the ordinary gamma function. We can also apply the recursive relation of the gamma function $\Gamma(a+1) = a\,\Gamma(a)$ and conclude that $1/c_{K_\Gamma} = (\pi b)^{d/2}\,\Gamma(d/2+a)/\Gamma(d/2+1)$.

Finally, we change the parametrization by setting $a = p/2, b = 2\sigma^2$ to obtain the final kernel:

$$\boxed{K_\Gamma(\boldsymbol{x}\,|\,p,\sigma^2) = \frac{1}{(2\pi\sigma^2)^{\frac{d}{2}}}\,\frac{\Gamma(\frac{d+2}{2})}{\Gamma(\frac{d+p}{2})}\,\Gamma(\tfrac{p}{2}, \tfrac{\|\boldsymbol{x}\|^2}{2\sigma^2})} \tag{10}$$

These *incomplete gamma kernels* span a family of Mean Shift kernels corresponding to L_p estimators of the attraction energy localized by a Gaussian kernel. An important special case of this family is the LOP kernel for which we choose $p = 1, \sigma^2 = 1/32$ and apply the identity $\Gamma(1/2, x) = \sqrt{\pi}\,\operatorname{erfc}(\sqrt{x})$ to get

$$K_{\mathrm{LOP}}(\boldsymbol{x}) = \frac{4^d}{\pi^{\frac{d-1}{2}}}\,\frac{\Gamma(\frac{d+2}{2})}{\Gamma(\frac{d+1}{2})}\,\operatorname{erfc}(4\,\|\boldsymbol{x}\|) \tag{11}$$

where erfc denotes the *complementary error function*. Another special case is the corresponding Gaussian kernel K_{Gaussian} obtained by setting $p = 2$ which is a common choice in Mean Shift and has been extensively analyzed as the localized L_2 estimator of the geometric mean. Fig. 2 shows an interpolation between these kernels by varying the p-norm.

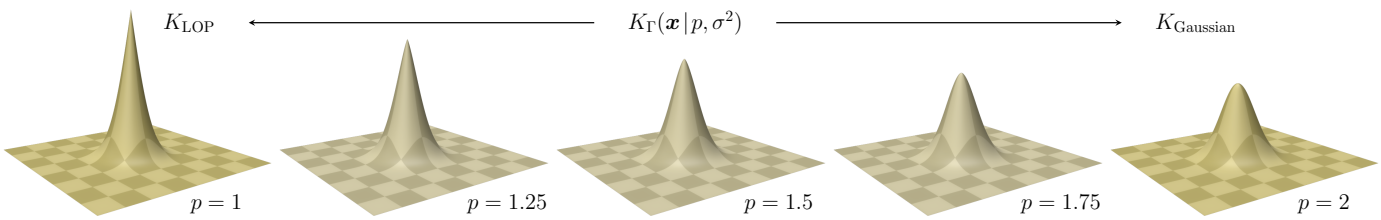


Fig. 2. Interpolation between 2D incomplete gamma kernels K_Γ with varying $p \in [1,2]$ and fixed $\sigma^2 = 1/32$. Each kernel corresponds to a localized attraction energy minimization with the respective p-norm.

TABLE 1
Properties of Incomplete Gamma Kernels $K_\Gamma(\boldsymbol{x}\,|\,p,\sigma^2)$ in $\mathbb{R}^d$ for $p>0$

Distribution	Mean	$\mathbf{0}$
	Covariance	$\frac{d+p}{d+2}\,\sigma^2\,\mathbf{I}$
	Characteristic function	${}_1\mathrm{F}_1(\frac{d+p}{2}, \frac{d+2}{2}, -\frac{\sigma^2\|\boldsymbol{\omega}\|^2}{2})$
	Moment-generating function	${}_1\mathrm{F}_1(\frac{d+p}{2}, \frac{d+2}{2}, \frac{\sigma^2\|\boldsymbol{\omega}\|^2}{2})$
Mean Shift	Differentiable profile	✓ except for $x=0$ if $p\in(0,2)$
	Strictly decreasing profile	✓
	Convex profile	✓ for $p\in(0,2]$
	Analytic	✓
	Bounded	✓
Other	Completely monotonic profile	✓ for $p\in(0,2]$
	Strictly positive definite	✓ for $p\in(0,2]$

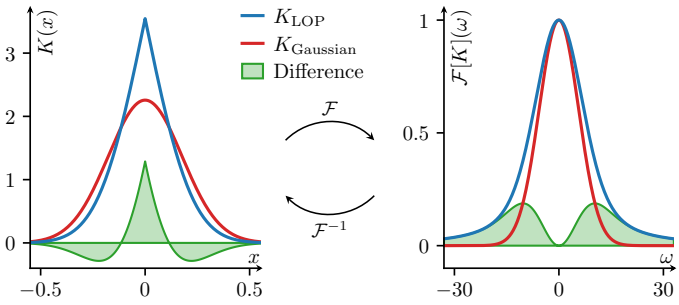


Fig. 3. Comparison of LOP and Gaussian kernels in spatial and frequency domain. Filtering with the LOP kernel K_{LOP} better preserves higher frequency information.

4 Kernel Properties

In the following, we derive several theoretical properties of the family of incomplete gamma kernels in $\mathbb{R}^d$ which are summarized in Table 1 and later leveraged in the applications (see Section 5).

4.1 Characteristic Function and Fourier Transform

Feature preservation is a highly relevant aspect in the development of denoising approaches (see Fig. 4) and will later be taken into account in the definition of density weighting schemes (see Section 5.1), loss functions (see Section 5.3), and neural network priors (see Section 5.4). In order to gain a deeper understanding of the proposed kernel family K_Γ in this context, we are interested in its characteristic function $\varphi_\Gamma(\boldsymbol{\omega})$ which can also be interpreted as the Fourier transform $\mathcal{F}$ of K_Γ at the angular frequency $\boldsymbol{\omega}\in\mathbb{R}^d$.

First, we can apply the relation between the d-dimensional Fourier transform of a radially symmetric function $f(\boldsymbol{x})$ in terms of the Hankel transform of order $d/2-1$ of the function $\|\boldsymbol{x}\|^{d/2-1}\,f(\|\boldsymbol{x}\|)$ [74] to reduce the dimensionality of the integral to the radial component

$$\begin{aligned}\varphi_\Gamma(\boldsymbol{\omega}) &= \mathcal{F}\big[K_\Gamma(\boldsymbol{x}\,|\,p,\sigma^2)\big](\boldsymbol{\omega}) = \int_{\mathbb{R}^d} K_\Gamma(\boldsymbol{x}\,|\,p,\sigma^2)\,\mathrm{e}^{\mathrm{i}\langle\boldsymbol{\omega}|\boldsymbol{x}\rangle}\,\mathrm{d}\boldsymbol{x}\\ &= c_{K_\Gamma}\frac{(2\pi)^{\frac{d}{2}}}{\|\boldsymbol{\omega}\|^{\frac{d}{2}-1}}\int_0^\infty \Gamma(\tfrac{p}{2},\tfrac{r^2}{2\sigma^2})\;\mathrm{J}_{\frac{d}{2}-1}(\|\boldsymbol{\omega}\|\,r)\,r^{\frac{d}{2}}\,\mathrm{d}r\end{aligned} \tag{12}$$

where $\mathrm{J}_q(x)$ denotes the *Bessel function of the first kind* of order q. This integral has the closed-form solution (see the Appendix for a more detailed derivation):

$$\begin{aligned}&c_{K_\Gamma}\frac{(2\pi)^{\frac{d}{2}}}{\|\boldsymbol{\omega}\|^{\frac{d}{2}-1}}\int_0^\infty \Gamma(\tfrac{p}{2},\tfrac{r^2}{2\sigma^2})\;\mathrm{J}_{\frac{d}{2}-1}(\|\boldsymbol{\omega}\|\,r)\,r^{\frac{d}{2}}\,\mathrm{d}r\\ &= c_{K_\Gamma}(2\pi\sigma^2)^{\frac{d}{2}}\frac{\Gamma(\frac{d+p}{2})}{\Gamma(\frac{d+2}{2})}\,{}_1\mathrm{F}_1(\tfrac{d+p}{2},\tfrac{d+2}{2},-\tfrac{\sigma^2\|\boldsymbol{\omega}\|^2}{2})\\ &= {}_1\mathrm{F}_1(\tfrac{d+p}{2},\tfrac{d+2}{2},-\tfrac{\sigma^2\|\boldsymbol{\omega}\|^2}{2})\end{aligned} \tag{13}$$

Therefore, the characteristic function of the incomplete gamma kernel can be written in terms of the *confluent hypergeometric function of the first kind* ${}_1\mathrm{F}_1$. Fig. 3 shows a comparison between the Gaussian kernel ($p=2$) and the LOP kernel ($p=1$) both in spatial and in frequency domain.

If we consider the special case ${}_1\mathrm{F}_1(a,a,x)=\mathrm{e}^x$, we can observe that this result is consistent with the Fourier transform of the Gaussian kernel. Furthermore as $d\to\infty$, the entire family of localized L_p kernel estimators converges to the L_2 estimator since distances become increasingly similar in higher dimensions due to the curse of dimensionality.

4.2 Moment-generating Function

Another closely related and useful quantity to consider is the moment-generating function M_Γ of the kernel K_Γ which can be used to compute its mean vector $\boldsymbol{\mu}_\Gamma\in\mathbb{R}^d$ and its covariance matrix $\boldsymbol{\Sigma}_\Gamma\in\mathbb{R}^{d\times d}$. Especially the insights about the magnitude of $\boldsymbol{\Sigma}_\Gamma$ are crucial to define consistent density weights (see Section 5.1) and an accurate CLOP approximation (see Section 5.2). Although there is a direct connection to the characteristic function in terms of

$$M_\Gamma(\boldsymbol{\omega}) = \varphi_\Gamma(-\mathrm{i}\,\boldsymbol{\omega}) = {}_1\mathrm{F}_1(\tfrac{d+p}{2},\tfrac{d+2}{2},\tfrac{\sigma^2\|\boldsymbol{\omega}\|^2}{2}) \tag{14}$$

it does not necessarily exist in general, so we have to prove this property for all $\boldsymbol{\omega}\in\mathbb{R}^d$. For this purpose, we repeatedly apply the comparison theorem of calculus to derive finite bounds of the integral. We first observe that $M_\Gamma(\boldsymbol{\omega})>0$ since the integrand consisting of K_Γ and an exponential term is positive. To obtain an upper bound of M_Γ, we switch to spherical coordinates and bound $\cos(\angle(\boldsymbol{\omega},\boldsymbol{x}))\le 1$, where $\angle(\boldsymbol{\omega},\boldsymbol{x})$ denotes the angle between $\boldsymbol{\omega}$ and $\boldsymbol{x}$, to decouple the radial component from the angular one:

$$\begin{aligned}M_\Gamma(\boldsymbol{\omega}) &= \int_{\mathbb{R}^d} K_\Gamma(\boldsymbol{x}\,|\,p,\sigma^2)\,\mathrm{e}^{\langle\boldsymbol{\omega}|\boldsymbol{x}\rangle}\,\mathrm{d}\boldsymbol{x}\\ &= c_{K_\Gamma}\int_\Omega\int_0^\infty \Gamma(\tfrac{p}{2},\tfrac{r^2}{2\sigma^2})\,r^{d-1}\,\mathrm{e}^{\|\boldsymbol{\omega}\|r\cos(\angle(\boldsymbol{\omega},\boldsymbol{x}))}\,\mathrm{d}r\,\mathrm{d}\boldsymbol{\Omega}\\ &\le c_1\int_0^\infty \Gamma(\tfrac{p}{2},\tfrac{r^2}{2\sigma^2})\,r^{d-1}\,\mathrm{e}^{\|\boldsymbol{\omega}\|r}\,\mathrm{d}r\end{aligned} \tag{15}$$

For brevity, we put finite terms into constants c_i. Next, we combine the two individual upper bounds

$$\Gamma(a,x)\le\begin{cases} a\,x^{a-1}\,\mathrm{e}^{-x}, & a\in[1,\infty), x\in[a,\infty)\ \text{[75]}\\ x^{a-1}\,\mathrm{e}^{-x}, & a\in(0,1], x\in(0,\infty)\ \text{(partial int.)}\end{cases} \tag{16}$$

and apply them after splitting the integral at $r_0 = \max(1, p/2)$. Furthermore, we simplify the expression by completing the square in the exponential term:

$$\begin{aligned} & c_1 \int_0^\infty \Gamma(\tfrac{p}{2}, \tfrac{r^2}{2\sigma^2})\, r^{d-1}\, \mathrm{e}^{\|\boldsymbol{\omega}\| r}\, \mathrm{d}r \\ & \leq c_1\, c_2 + c_1 \int_{r_0}^\infty r_0\, (\tfrac{r^2}{2\sigma^2})^{\frac{p}{2}-1}\, \mathrm{e}^{-\frac{r^2}{2\sigma^2}}\, r^{d-1}\, \mathrm{e}^{\|\boldsymbol{\omega}\| r}\, \mathrm{d}r \\ & = c_1\, c_2 + c_1\, c_3 \int_{r_0}^\infty r^{p+d-3}\, \mathrm{e}^{-\frac{(r-\sigma^2\|\boldsymbol{\omega}\|)^2}{2\sigma^2}}\, \mathrm{d}r \end{aligned} \tag{17}$$

Since $r \in [1, \infty)$, the remaining polynomial can be bound by a higher order $k = \max(0, \lceil p + d - 3 \rceil) \in \mathbb{N}_0$. Therefore, this integral describes the (incomplete) k-th raw moment of a 1D normal distribution and is finite for any $k \in \mathbb{N}_0$ which, in turn, proves that $M_\Gamma(\boldsymbol{\omega}) < \infty$ exists.

4.2.1 Mean

We can now use the moment-generating function M_Γ to directly compute all raw moments of the kernel K_Γ by evaluating the respective derivative at $\boldsymbol{\omega} = \mathbf{0}$. For the mean vector $\boldsymbol{\mu}_\Gamma \in \mathbb{R}^d$ of K_Γ, we consider the first-order derivative

$$\frac{\partial}{\partial \boldsymbol{\omega}} M_\Gamma(\boldsymbol{\omega}) = \tfrac{d+p}{d+2}\, \sigma^2\, {}_1\mathrm{F}_1(\tfrac{d+p+2}{2}, \tfrac{d+4}{2}, \tfrac{\sigma^2\|\boldsymbol{\omega}\|^2}{2})\, \boldsymbol{\omega} \tag{18}$$

and get $\boldsymbol{\mu}_\Gamma = \frac{\partial}{\partial \boldsymbol{\omega}} M_\Gamma(\mathbf{0}) = \mathbf{0}$ as the expected result for a radially symmetric kernel.

4.2.2 Covariance

Similarly, we compute the second-order derivative

$$\begin{aligned} \frac{\partial^2}{\partial \boldsymbol{\omega}\, \partial \boldsymbol{\omega}^\mathsf{T}} M_\Gamma(\boldsymbol{\omega}) = \tfrac{d+p}{d+2}\, \sigma^2 \Big[& {}_1\mathrm{F}_1(\tfrac{d+p+2}{2}, \tfrac{d+4}{2}, \tfrac{\sigma^2\|\boldsymbol{\omega}\|^2}{2})\, \mathbf{I} \\ & + \tfrac{d+p+2}{d+4}\, \sigma^2\, {}_1\mathrm{F}_1(\tfrac{d+p+4}{2}, \tfrac{d+6}{2}, \tfrac{\sigma^2\|\boldsymbol{\omega}\|^2}{2})\, \boldsymbol{\omega}\boldsymbol{\omega}^\mathsf{T} \Big] \end{aligned} \tag{19}$$

and obtain the covariance matrix $\boldsymbol{\Sigma}_\Gamma \in \mathbb{R}^{d \times d}$ of the kernel K_Γ using the first-order and second-order raw moments as $\boldsymbol{\Sigma}_\Gamma = \frac{\partial^2}{\partial \boldsymbol{\omega}\, \partial \boldsymbol{\omega}^\mathsf{T}} M_\Gamma(\mathbf{0}) - \boldsymbol{\mu}_\Gamma \boldsymbol{\mu}_\Gamma^\mathsf{T} = [(d+p)/(d+2)]\, \sigma^2\, \mathbf{I}$.

4.3 Mean Shift Properties

In addition to the distribution-specific properties above, we can get further insights into the kernel family by exploiting the comprehensive theory that has been developed for the Mean Shift algorithm [70]. This requires proving several additional properties including that the kernel K_Γ is bounded and analytic and that its profile k_Γ is differentiable, strictly decreasing, and convex.

4.3.1 Differentiability, Monotonicity and Convexity

In order to show that the profile is strictly decreasing, we consider its first-order derivative

$$\frac{\mathrm{d}}{\mathrm{d}x} k_\Gamma(x\,|\,p, \sigma^2) = \frac{\mathrm{d}}{\mathrm{d}x} \Gamma(\tfrac{p}{2}, \tfrac{x}{2\sigma^2}) = -(\tfrac{1}{2\sigma^2})^{\frac{p}{2}}\, x^{\frac{p}{2}-1}\, \mathrm{e}^{-\frac{x}{2\sigma^2}} \tag{20}$$

which is defined for all $x \in (0, \infty)$ as well as for $x = 0$ if $p \in [2, \infty)$. Since the involved polynomial and exponential terms are always positive, it follows that the derivative must be negative, that is $\frac{\mathrm{d}}{\mathrm{d}x} k_\Gamma(x\,|\,p, \sigma^2) < 0$, and the profile strictly decreasing.

Similarly, we see that the second-order derivative is given by

$$\frac{\mathrm{d}^2}{\mathrm{d}x^2} k_\Gamma(x\,|\,p, \sigma^2) = \frac{\mathrm{d}}{\mathrm{d}x} k_\Gamma(x\,|\,p, \sigma^2) \left[\frac{\frac{p}{2} - 1}{x} - \frac{1}{2\sigma^2} \right] \tag{21}$$

where, in order to ensure that $\frac{\mathrm{d}^2}{\mathrm{d}x^2} k_\Gamma(x\,|\,p, \sigma^2) > 0$, the latter term must be non-positive which is equivalent to the condition $x \geq (p-2)\, \sigma^2$. Since this should hold for all $x \in (0, \infty)$, convexity is only guaranteed for kernels with $p \in (0, 2]$ which, in particular, includes the Gaussian kernel ($p = 2$) as well as the LOP kernel ($p = 1$).

4.3.2 Boundedness and Analyticity

Instead of showing the properties of boundedness and analyticity for the kernel K_Γ itself, it is sufficient to show them for its profile k_Γ. Since k_Γ is non-negative and monotonically decreasing, we only have to consider the case $x = 0$. For this value, $\Gamma(a, x)$ reduces to the gamma function $\Gamma(a)$ which is finite for $a > 0$. Furthermore, analyticity directly follows from the fact that $\Gamma(a, x)$ is holomorphic in $x \in (0, \infty)$ for any fixed $a > 0$.

4.3.3 Consequences for the LOP operator

The aforementioned properties have several direct implications [70] on the behavior of the LOP operator (with zero repulsion) as well as to Mean Shift applied with the incomplete gamma kernel K_Γ for $p \in (0, 2]$. With the exception of the finite set of target points $\mathcal{P}$ where singularities are introduced in the kernel G_Γ, the following properties hold:

Non-zero Gradient. The gradient of the kernel density estimate $\nabla \hat{f}_{\mathcal{P}, K_\Gamma}$ in (2) is non-zero outside the convex hull of the target point set $\mathcal{P}$. This implies that all solutions must lie within the convex hull.

Plateau-free Density. In addition to non-zero gradients, the kernel density estimate on the set $\mathbb{R}^d \setminus \mathcal{P}$ has no plateaus. Since the set of target points $\mathcal{P}$ is finite, we can extend this property to the full space $\mathbb{R}^d$.

Non-decreasing Density Estimate. Another interesting subset to consider is the improvement ball $\mathcal{I}(\boldsymbol{q}_j^{(t)})$ which denotes a d-dimensional sphere centered at the point $\boldsymbol{q}_j^{(t)} + \boldsymbol{m}_{\mathcal{P}, G_\Gamma}(\boldsymbol{q}_j^{(t)})$ with radius $\|\boldsymbol{m}_{\mathcal{P}, G_\Gamma}(\boldsymbol{q}_j^{(t)})\|$. In case of the LOP operator, it follows that all points $\boldsymbol{x}$ within the improvement ball have non-decreasing kernel density estimates $\hat{f}_{\mathcal{P}, K_\Gamma}(\boldsymbol{x}) \geq \hat{f}_{\mathcal{P}, K_\Gamma}(\boldsymbol{q}_j^{(t)})$.

Convergence of Density Estimate Sequence. As a consequence of the above property, the sequence of kernel density estimates $\{\hat{f}_{\mathcal{P}, K_\Gamma}(\boldsymbol{q}^{(t)})\}$ obtained via the fixed-point iteration $\boldsymbol{q}^{(t+1)} = \boldsymbol{q}^{(t)} + \boldsymbol{m}_{\mathcal{P}, G_\Gamma}(\boldsymbol{q}^{(t)})$ is non-decreasing. Furthermore, this sequence always converges.

Convergence of Mode Estimate Sequence. Finally, we can conclude that the mode estimate sequence $\{\boldsymbol{q}^{(t)}\}$ converges to a single point. Depending on the window size h and the distribution of the target points $\mathcal{P}$, this solution could be either a point $\boldsymbol{p} \in \mathcal{P}$ due to the singularity (for very small window sizes) or a different point $\boldsymbol{p} \in \mathbb{R}^d \setminus \mathcal{P}$ in the corresponding convex hull (for larger window sizes). While a proof for the convergence of a modified version of LOP in $\mathbb{R}^d$ has only been shown very recently [76], this provides an alternative way through the comprehensive Mean Shift theory which, though only considering the attraction term, extends and generalizes to all kernels K_Γ with $p \in (0, 2]$.

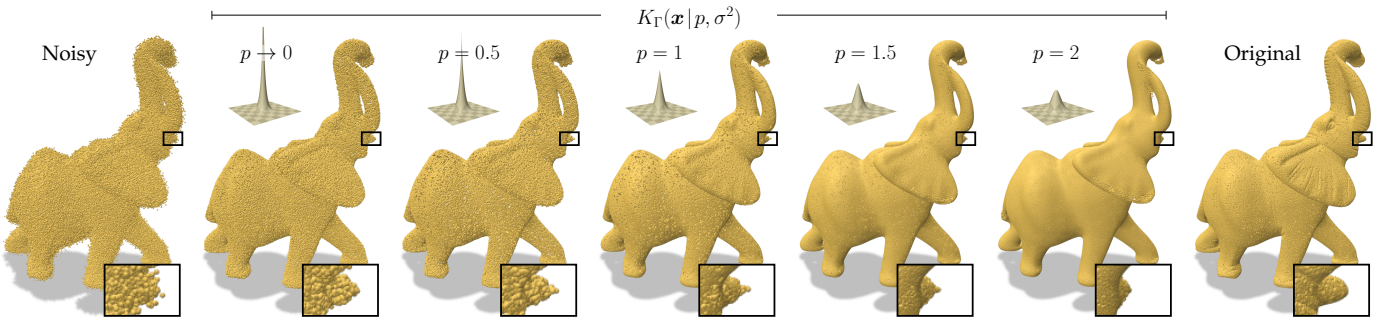


Fig. 4. Exemplary point cloud denoising of the *Elephant* model (302 458 points) with 30 WLOP [2] iterations ($h = 6$, $\mu = 0.4$) for different incomplete gamma kernels K_Γ using varying $p \in (0, 2]$ and fixed $\sigma^2 = 1/32$. The model has been corrupted with $\sigma_{\text{noise}} = 0.3$ (80 % points) and $\sigma_{\text{outlier}} = 1.5$ Gaussian noise (20 % points) respectively to account for both typical sensor noise and heavy outliers. Higher p-norms result in more regular but oversmoothed point distributions whereas lower values better preserve features. Unit of $h, \sigma_{\text{noise}}, \sigma_{\text{outlier}}$: [% BB diagonal].

4.4 Further Properties

Besides the theoretical results concerning basic distribution-related aspects as well as insights in the projection behavior of LOP from the perspective of Mean Shift, we want to derive further properties and, in particular, strict positive definiteness. This opens up a broader set of applications beyond improved density weights (see Section 5.1) and may also be relevant, e.g., in the field of Gaussian Process Regression, which, however is out of the scope of this work.

4.4.1 Complete Monotonicity

For this purpose, we show that the kernel profile k_Γ is completely monotonic, that is $(-1)^n \frac{\mathrm{d}^n}{\mathrm{d}x^n} k_\Gamma(x\,|\,p,\sigma^2) \geq 0$ for all $n \in \mathbb{N}_0$ and $x \in (0, \infty)$. From the derivation of the Mean Shift properties, we already know that $k_\Gamma(x\,|\,p,\sigma^2) > 0$ and $\frac{\mathrm{d}}{\mathrm{d}x} k_\Gamma(x\,|\,p,\sigma^2) < 0$ holds for all $x \in (0, \infty)$. Since x^a with $a \leq 0$ as well as e^{-x} are both completely monotonic and the product of two completely monotonic functions retains that property [77], we can conclude that this also holds for $k_\Gamma(x\,|\,p,\sigma^2)$ with $p \in (0, 2]$.

4.4.2 Strict Positive Definiteness

A direct consequence of the complete monotonicity of its profile is that the kernel K_Γ is a positive definite function. Furthermore, since the profile is also not a constant function, K_Γ must be even strictly postive definite [78]. Therefore, for any set of distinct points $\mathcal{P}$, the matrix

$$\boldsymbol{C} = \left(K_\Gamma(\boldsymbol{p}_i - \boldsymbol{p}_j\,|\,p,\sigma^2)\right)_{ij} \in \mathbb{R}^{|\mathcal{P}|\times|\mathcal{P}|} \tag{22}$$

is symmetric and positive definite for $p \in (0, 2]$, so it can be interpreted as a Gram matrix and any linear system with respect to $\boldsymbol{C}$ has a unique solution which can be computed by, e.g., conjugate gradient solvers. This also directly extends to any truncated version of K_Γ where the matrix $\boldsymbol{C}$ becomes sparse and more efficient to solve as vanishing derivatives of the truncated profile do not affect complete monotonicity.

5 Applications

Besides the application of other localized L_p estimators for point cloud denoising via the incomplete gamma kernels K_Γ, as shown in Fig. 4, we illustrate several further applications to demonstrate the benefits of the theoretical results derived for the kernel family.

5.1 WLOP Density Weights

In addition to a high robustness to noise, another crucial requirement in point cloud denoising is a uniform distribution of the denoised points. Although the introduction of an additional repulsion energy (6) mitigates the clustering effect of the attraction term (5), the projection is still highly dependent on the distribution of the target points $\mathcal{P}$. This has been addressed in WLOP [2] by computing weights v_i for each target point $\boldsymbol{p}_i$ and $v_j^{(t)}$ for each projection point $\boldsymbol{q}_j^{(t)}$ as an additional regularization based on the reciprocal and ordinary density value respectively with the (unnormalized) localization kernel θ. However, our derived theoretical properties reveal two major limitations of this particular choice: 1) Although the Gaussian localization kernel θ could be considered a reasonable approximation of the actual kernel K_{LOP} given in (11), it significantly differs in terms of the frequency spectrum (see Section 4.1 and Fig. 3) as well as covariance (see Section 4.2) and leads to oversmoothing of high-frequency density information and, hence, a lacking preservation of fine-scale details; 2) taking the reciprocal to invert the density of $\boldsymbol{p}_i$ ignores the dependencies between the weights which corresponds to the assumption of constant density in a window of size h. In order to achieve a more accurate normalization, we propose two novel weighting schemes.

Simple Scheme. A simple extension to the WLOP weights keeps the assumption of the latter limitation and addresses only the former one by applying the actual kernel K_{LOP}, that is estimating the weights

$$v_i = \frac{1}{\hat{f}_{\mathcal{P},K_{\text{LOP}}}(\boldsymbol{p}_i)}, \quad v_j^{(t)} = \hat{f}_{\mathcal{Q}^{(t)},K_{\text{LOP}}}(\boldsymbol{q}_j^{(t)}) \tag{23}$$

via the kernel density estimate (1) of the point clouds $\mathcal{P}$ and $\mathcal{Q}^{(t)}$ respectively. This scheme can be easily integrated into existing applications of WLOP as it only involves a different kernel function in the overall weight computation.

Full Scheme. To address both limitations, we consider the kernel density estimate $\hat{f}_{\mathcal{P},K_{\text{LOP}}}$ with weights v_i applied to each term. We want to enforce constant density at the points $\boldsymbol{p}_i$ which can be formulated as a linear optimization problem in matrix form:

$$\frac{1}{|\mathcal{P}|\,h^d}\left(K_{\text{LOP}}(\tfrac{\boldsymbol{p}_i - \boldsymbol{p}_j}{h})\right)_{ij}(\dots, v_i, \dots)^{\mathsf{T}} = \boldsymbol{1} \tag{24}$$

TABLE 2
Parameter Sets of Kernel Approximation $\hat{K}_{\mathrm{LOP}}$

	CLOP [5]		Ours		Ours (Consistent)	
k	$\hat{w}_k$	$\hat{\sigma}_k$	$\hat{w}_k$	$\hat{\sigma}_k$	$\hat{w}_k$	$\hat{\sigma}_k$
1	97.761	0.01010	61.509	0.02102	46.409	0.03118
2	29.886	0.03287	11.932	0.07289	9.635	0.10582
3	11.453	0.11772	5.069	0.15700	2.674	$\sqrt{1/32}$

This corresponds to radial basis function (RBF) interpolation with a Gram matrix similar to (22) and we can obtain a unique solution since K_{LOP} is strictly positive definite (see Section 4.4). Furthermore, we truncate the kernel at $h/2$ to drastically reduce the memory requirements of the matrix and use a sparse conjugate gradient solver. In case of the projection points $\boldsymbol{q}_j^{(t)}$, we consider the inverse of the matrix which leads to the same weights as in the simple scheme.

5.2 CLOP Kernel Approximation

While the major aspect of ensuring a uniform distribution of the projected point set can be addressed with an appropriate weighting as shown in Section 5.1, obtaining these results efficiently becomes challenging especially on large datasets as all target points $\mathcal{P}$ will be taken into account for denoising. For this purpose, CLOP [5] first computes a more compact representation of the points $\mathcal{P}$ in terms of a smaller set of normal distributions $\mathcal{P}_{\mathcal{N}} = \{(w_i, \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)\}$ with weights $w_i \in \mathbb{R}$, mean vectors $\boldsymbol{\mu}_i \in \mathbb{R}^3$, and local covariance matrices $\boldsymbol{\Sigma}_i \in \mathbb{R}^{3\times 3}$ and then extends the discrete attraction energy to the continuous space. However, since the integral in the respective update step cannot be directly solved, the kernel $\alpha(\|\boldsymbol{x}\|)$ is approximated by a radially symmetric Gaussian mixture model $\hat{\alpha}(\boldsymbol{x}) = (1/h)\sum_{k=1}^{3} \hat{w}_k\, \hat{c}_k\, \mathcal{N}(\boldsymbol{x}/h \,|\, \mathbf{0}, \hat{\sigma}_k^2\, \mathbf{I})$ consisting of three components with fitted parameters $\{(\hat{w}_k, \hat{\sigma}_k) \in \mathbb{R} \times \mathbb{R}\}$ and dimension-dependent constants $\hat{c}_k = |2\pi\hat{\sigma}_k^2\, \mathbf{I}|^{1/2}$. In the context of Mean Shift, this implies that the kernel G_{LOP} is in fact approximated which directly allows us to derive

$$\hat{K}_{\mathrm{LOP}}(\boldsymbol{x}) = \frac{\sum_{k=1}^{3} \hat{\sigma}_k^2\, \hat{w}_k\, \hat{c}_k\, \mathcal{N}(\boldsymbol{x} \,|\, \mathbf{0}, \hat{\sigma}_k^2\, \mathbf{I})}{\sum_{k=1}^{3} \hat{\sigma}_k^2\, \hat{w}_k\, \hat{c}_k} \tag{25}$$

as an approximation of the kernel K_{LOP} in $\mathbb{R}^d$ with the same set of fitted parameters $\{(\hat{w}_k, \hat{\sigma}_k)\}$.

Kernel Fit. Finding the optimal parameter set is highly challenging due to the singularity of G_{LOP} at $x = 0$ and, thereby, the unbounded ratio between the smallest and largest sampling value in the half-open fitting interval $(0, 1]$ for $h = 1$. In contrast, we directly optimize on the kernel K_{LOP} which does not suffer from these limitations. We fix the parameter $w_3 = 1$ to constrain the remaining degree of freedom and obtain the solution from 10^7 uniformly sampled points in the interval $[0, 1]$ via the Levenberg-Marquardt algorithm (see Table 2). Although the LOP operator is scale-invariant in terms of the kernel α, we nevertheless estimate a global scaling factor for the weights $\hat{w}_k$ via Levenberg-Marquardt optimization in the interval $(0.01, 1]$ for a better comparability with CLOP.

Consistent Fit. In addition to the derivation of the kernel approximation $\hat{K}_{\mathrm{LOP}}$, we can futher apply the insights about the covariance of K_{LOP} (see Section 4.2). From (25), we can also see that the covariance matrix $\hat{\boldsymbol{\Sigma}}_{\mathrm{LOP}} \in \mathbb{R}^{d\times d}$ of the approximation consists of a convex combination of the parameters $\hat{\sigma}_k^2$:

$$\hat{\boldsymbol{\Sigma}}_{\mathrm{LOP}} = \frac{\sum_{k=1}^{3} \hat{w}_k\, \hat{\sigma}_k^{d+4}}{\sum_{k=1}^{3} \hat{w}_k\, \hat{\sigma}_k^{d+2}}\, \mathbf{I} \tag{26}$$

In the limit $d \to \infty$, this combination degenerates to $\hat{\boldsymbol{\Sigma}}_{\mathrm{LOP}} \to \max_k \hat{\sigma}_k^2\, \mathbf{I}$ which is similar to the maximum norm L_∞ being the limit of the L_p norms. Therefore, we can enforce an additional consistency constraint in the parameter optimization process by fixing the parameter $\hat{\sigma}_3 = \sqrt{1/32}$ such that $\hat{\boldsymbol{\Sigma}}_{\mathrm{LOP}}$ matches the expected covariance $\boldsymbol{\Sigma}_{\mathrm{LOP}} = 1/32\,[(d+1)/(d+2)]\,\mathbf{I} \to 1/32\,\mathbf{I}$.

5.3 Robust Loss Functions

A closely related, yet slightly different task is mesh denoising where the quality of the noisy vertex set (similar to point sets in point cloud denoising) should be improved while preserving the mesh connectivity and avoiding self-intersections of the faces. Respective approaches tackle this problem in a two-stage approach where the face normals are initially denoised and subsequently used as a guidance in the second stage to consistently adjust the vertex positions. Obtaining a reliable estimate of a denoised normal $\boldsymbol{n} \in \mathbb{R}^3$ can be performed by leveraging gradient descent or M-estimation from the broad field of robust statistics [79] which is also related to the concept of anisotropic diffusion [80]. Here, the objective function

$$L(\boldsymbol{n}) = \sum_i \rho(\|\boldsymbol{n}_i - \boldsymbol{n}\|) \tag{27}$$

defined with a robust loss function ρ is considered and a solution can be found based on the corresponding influence function $\Psi(x) = \frac{\mathrm{d}}{\mathrm{d}x}\rho(x)$ and anisotropic weight function $\tilde{g}(x) = \Psi(x)/x$ [81]:

$$\boldsymbol{n}^{(t+1)} = \frac{\sum_i \tilde{g}(\|\boldsymbol{n}_i - \boldsymbol{n}^{(t)}\|)\, \boldsymbol{n}^{(t)}}{\left\|\sum_i \tilde{g}(\|\boldsymbol{n}_i - \boldsymbol{n}^{(t)}\|)\, \boldsymbol{n}^{(t)}\right\|} \tag{28}$$

This result is closely related to the derivation of Mean Shift and shares many properties with it [10]. While the Gaussian loss is a well-known choice in this context, we can generalize it to the family of *incomplete gamma losses* along with the respective influence and anisotropic weight

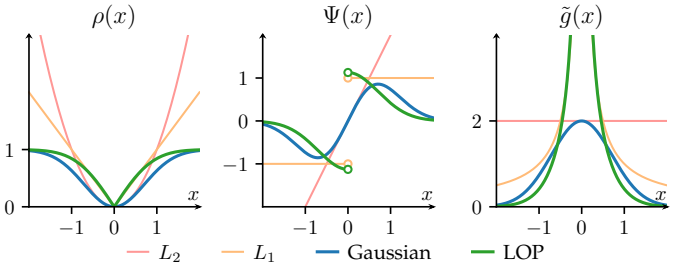


Fig. 5. Comparison of LOP and Gaussian M-estimators for $\sigma^2 = 1/2$. Due to the close relation to Mean Shift, these robust loss functions ρ do not only share the shape of the corresponding kernels K but also have similar properties and form localized versions of the common global L_2 and L_1 loss functions.

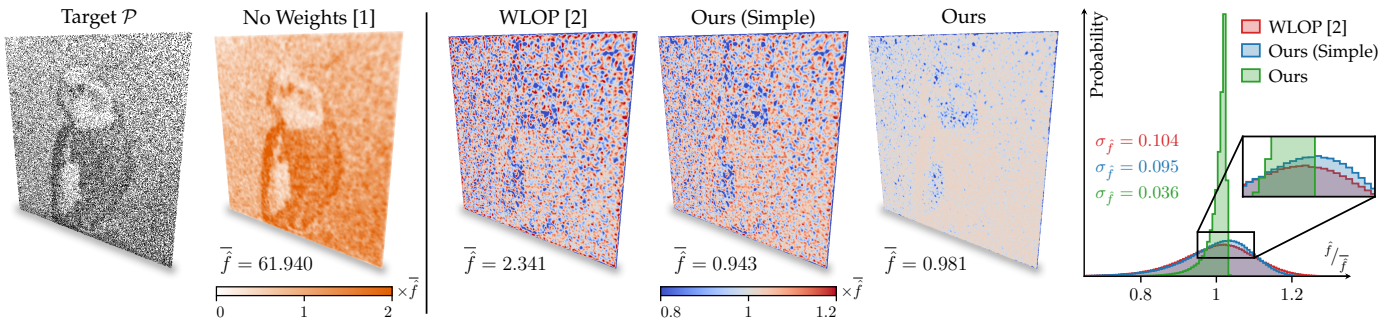


Fig. 6. Kernel density estimate $\hat{f}$ for $h = 3$ of a planar target surface patch $\mathcal{P}$ (74 000 points) that is sampled inversely proportional to the intensity of the *Bird* image. The regularity $\sigma_{\mathcal{Q}}$ of any projected point set $\mathcal{Q}$ onto this target directly depends on the uniformity of $\hat{f}$. Whereas WLOP [2] and our simple weighting scheme cannot fully remove high-frequency variations, our full weighting scheme leads to a significantly better normalization and more uniform density. Unit of h: [% BB diagonal].

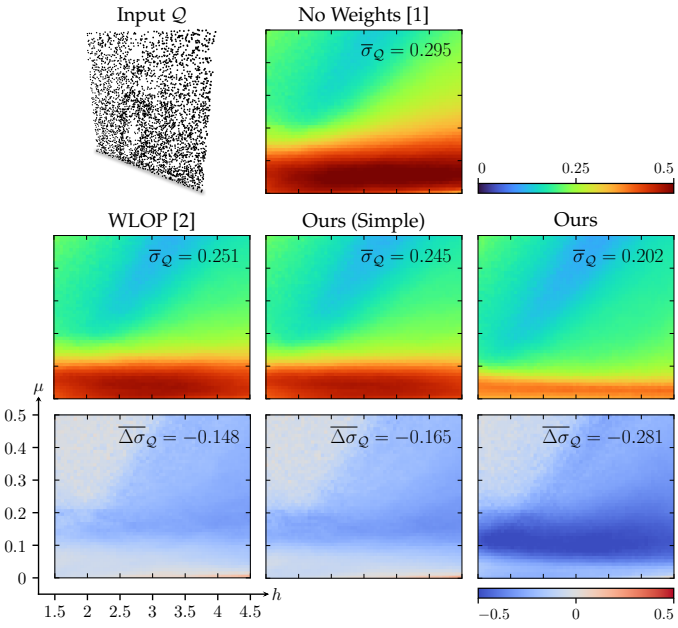


Fig. 7. Regularity $\sigma_{\mathcal{Q}}$ of the input subset $\mathcal{Q}$ (3700 points) projected on a planar target surface patch $\mathcal{P}$ (74 000 points) that is sampled inversely proportional to the intensity of the *Bird* image. Due to the better density normalization, our weighting schemes further improve the regularity across various combinations of the window size h and the repulsion weight μ. Unit of h and $\sigma_{\mathcal{Q}}$: [% BB diagonal].

functions and corresponding frequency-related properties (see Section 4.1):

$$\rho_\Gamma(x\,|\,p,\sigma^2) = \frac{1}{\Gamma(\frac{p}{2})}\,\gamma(\tfrac{p}{2},\tfrac{x^2}{2\sigma^2}) \tag{29}$$

$$\Psi_\Gamma(x\,|\,p,\sigma^2) = \frac{2}{(2\sigma^2)^{\frac{p}{2}}\,\Gamma(\frac{p}{2})}\,|x|^{p-2}\,\mathrm{e}^{-\frac{x^2}{2\sigma^2}}\,x \tag{30}$$

$$\tilde{g}_\Gamma(x\,|\,p,\sigma^2) = \frac{2}{(2\sigma^2)^{\frac{p}{2}}\,\Gamma(\frac{p}{2})}\,|x|^{p-2}\,\mathrm{e}^{-\frac{x^2}{2\sigma^2}} \tag{31}$$

Here, the losses ρ_Γ are built upon the *lower incomplete gamma function* $\gamma(a,x) = \int_0^x t^{a-1}\,\mathrm{e}^{-t}\,\mathrm{d}t$ which is connected to the upper incomplete gamma function via the relation $\gamma(a,x) + \Gamma(a,x) = \Gamma(a)$. By choosing $p = 1$ and applying the identity $\gamma(1/2,x) = \sqrt{\pi}\,\operatorname{erf}(\sqrt{x})$, we get the LOP loss

$$\rho_{\mathrm{LOP}}(x\,|\,\sigma^2) = \operatorname{erf}(\tfrac{|x|}{\sqrt{2\sigma^2}}) \tag{32}$$

where erf denotes the *error function* and is related to its complementary counterpart via $\operatorname{erfc}(x) = 1 - \operatorname{erf}(x)$. Considering $\sigma^2 = 1/32$, the relation $\tilde{g}_{\mathrm{LOP}}(x\,|\,1/32) \propto g_{\mathrm{LOP}}(x^2)$ further highlights the close similarity to Mean Shift. Due to the broad applicability of the concepts of robust statistics [79], i.e. the definition and optimization of loss functions, to traditional approaches, machine learning, and many other fields such as Mean Shift, our losses ρ_Γ may also be highly relevant for deep learning techniques and extend previous generalizations [82]. Fig. 5 shows a comparison between the Gaussian M-estimator ($p = 2$) and the LOP M-estimator ($p = 1$).

5.4 Neural Network Priors

In the context of learning-based point cloud denoising, many approaches rely on clean groundtruth data in order to learn how various types of synthetically generated noise such as Gaussian, uniform, or simulated sensor noise can be effectively removed from the given point set. However, they may not necessarily generalize to real-world data exhibiting different possibly systematic yet unknown noise patterns and, hence, lead to less reliable and accurate results. For this purpose, TotalDenoising [50] formulates the learning objective in an unsupervised manner, which only requires noisy input data without further knowledge about the noise itself or corresponding clean data. Given noisy points $\boldsymbol{p} \in \mathbb{R}^3$, the parameters $\boldsymbol{\Theta}$ of a neural network $f_{\boldsymbol{\Theta}}$ are optimized such that the observations $\boldsymbol{p}$ are projected to the unknown surface $\mathcal{S}$ represented by the distribution of modes $p(\boldsymbol{x}\,|\,\mathcal{S})$. In particular, this can be formulated by the objective function

$$L(\boldsymbol{\Theta}) = \mathbb{E}_{\boldsymbol{p}\sim p(\boldsymbol{x}\,|\,\mathcal{S})}\,\mathbb{E}_{\boldsymbol{q}\sim q(\boldsymbol{x}\,|\,\boldsymbol{p})}\,\rho(f_{\boldsymbol{\Theta}}(\boldsymbol{p},\boldsymbol{q})) \tag{33}$$

where ρ denotes a loss function, e.g. the commonly used L_2 loss or other choices such as robust losses (see Section 5.3), and $\boldsymbol{q}$ denotes the generated samples during training. Since any point $\boldsymbol{x} \in \mathcal{S}$ on the surface is a valid target onto which the noisy input $\boldsymbol{p}$ can be projected, the unsupervised training procedure will not converge to a unique solution. Thus, the prior distribution

$$q(\boldsymbol{x}\,|\,\boldsymbol{p}) = p(\boldsymbol{x}\,|\,\mathcal{S})\,K(\boldsymbol{W}(\boldsymbol{x}-\boldsymbol{p})) \tag{34}$$

regularizes this problem by convolving the distribution $p(\boldsymbol{x}\,|\,\mathcal{S})$ with a smoothing kernel K and a diagonal weight matrix $\boldsymbol{W} \in \mathbb{R}^{3\times3}$ to favor the closest mode as the unique

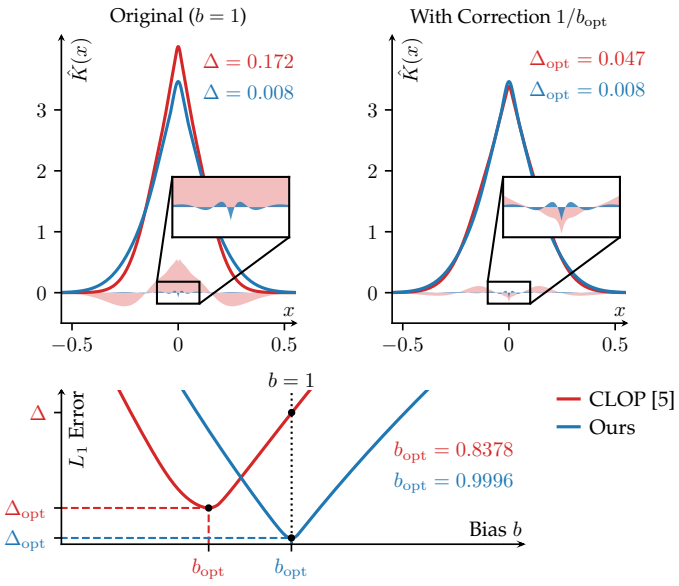


Fig. 8. Analysis of bias in the width of the kernel approximations $\hat{K}_{\text{LOP}}$ to the original kernel K_{LOP}. A correction by scaling the parameters $\hat{\sigma}_k$ with a global factor $1/b_{\text{opt}}$ lowers the error for CLOP [5]. Nevertheless, our approximation still better follows K_{LOP} with a significantly lower error and is almost unbiased in the 1-dimensional case.

TABLE 3
Ratio of Standard Deviations $\|\hat{\boldsymbol{\Sigma}}_{\text{LOP}}\|^{1/2}/\|\boldsymbol{\Sigma}_{\text{LOP}}\|^{1/2}$ Between the Kernel Approximations $\hat{K}_{\text{LOP}}$ and the Original Kernel K_{LOP} in $\mathbb{R}^d$

	$d=1$	$d=2$	$d=3$	$d=4$	$d\to\infty$
CLOP [5]	0.7931	0.7632	0.7430	0.7291	0.6659
Ours	**0.9917**	**0.9834**	**0.9737**	**0.9640**	0.8881
Ours (Consistent)	1.0137	1.0252	1.0362	1.0440	**1**

solution for projection. This convolution-based regularization is similar to the formulation of the kernel density estimate (1) in the Mean Shift approach. Therefore, instead of using a Gaussian kernel as in the original TotalDensoising approach, we can apply the gained insights of our kernel family K_Γ regarding feature preservation (see Section 4.1 and Fig. 3) and use the LOP kernel K_{LOP} given in (11) in the prior distribution.

6 Experimental Results

In the following, we demonstrate the effectiveness of our proposed extensions in Section 5 that are derived from the theoretical properties of the kernel family (see Section 4).

6.1 Evaluation of WLOP Density Weights

In order to evaluate the performance of our density weighting schemes defined in (23) and (24) (see Section 5.1), we measured the regularity of the point cloud $\mathcal{Q}$ after projection onto a highly irregular target $\mathcal{P}$ [2]. For this purpose, we sampled 74 000 target points from a 3D surface patch inversely proportional to the intensity of the mapped *Bird* image and took a random subset of 3700 points for projection. Then, we applied 100 iterations of the LOP operator as well as its weighted versions and computed the regularity

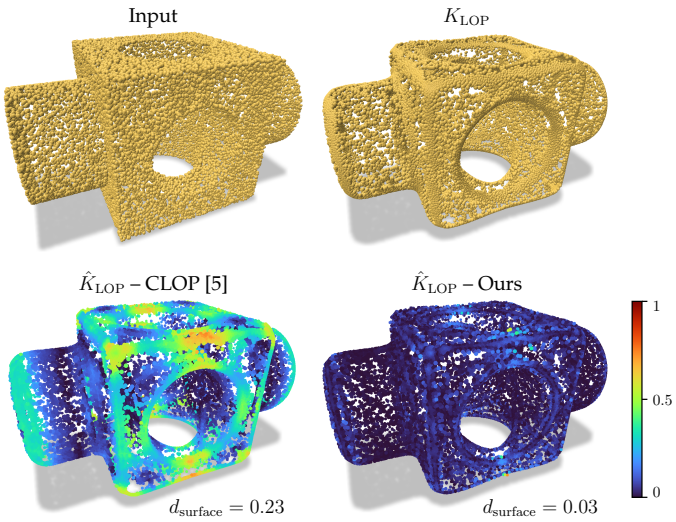


Fig. 9. Bias of the kernel approximations $\hat{K}_{\text{LOP}}$ when applying WLOP [2] to smooth the *Block* model (25 000 points) with the window size $h = 25$. Whereas the CLOP [5] approximation introduces systematic errors at the edges due to the bias in the width, our variant closely resembles the behavior of the original kernel K_{LOP}. Unit: [% BB diagonal].

$$\sigma_{\mathcal{Q}} = \left[\frac{1}{|\mathcal{Q}|}\sum_i \Big(d(\boldsymbol{q}_i, \mathcal{Q}\setminus\{\boldsymbol{q}_i\}) - \overline{d(\boldsymbol{q}_i, \mathcal{Q}\setminus\{\boldsymbol{q}_i\})}\Big)^2\right]^{\frac{1}{2}} \tag{35}$$

which is defined as the standard deviation of the nearest neighbor distances $d(\boldsymbol{x}, \mathcal{Y}) = \min_j \|\boldsymbol{x} - \boldsymbol{y}_j\|$ within the point cloud $\mathcal{Q}$. Fig. 7 shows the quantitative results for 60×50 combinations of h and μ. Throughout 76.7 % of all combinations, our simple weighting scheme (23) performs better than WLOP with a slightly lower value of $\sigma_{\mathcal{Q}}$ on average. Our full scheme (24) outperforms WLOP in 99.3 % and the simple scheme in 98.7 % of all combinations, especially in configurations with low repulsion weights $\mu \in [0, 0.2]$.

These improvements in point cloud regularity directly correspond to a more evenly distributed density along the surface. Fig. 6 depicts a comparison of 1000×1000 evenly sampled density values on the respective 3D surface patch. Both WLOP and our simple scheme (23) normalize the lower frequency components of the density, but still retain high-frequency variations due to the independent computation of each weight. On the other hand, our full scheme (24) does not suffer from these artifacts and only leads to underestimated densities at the boundary and in sparsely sampled regions where the window size h is not sufficiently large to bridge these gaps.

Since Mean Shift and, thereby, LOP and its variants are scale-invariant with respect to a global normalization constant, we computed the mean density $\overline{\hat{f}}$ for each weighting scheme and used this value to normalize each density distribution for a fair comparison. Whereas this value is close to one for both of our schemes due to the correct handling of the normalization constants, we can derive a theoretical estimate of this value for WLOP

$$\overline{\hat{f}}_{\text{WLOP}} \approx \frac{1}{|\mathcal{P}|\,h^3}\, c_\theta^{(d=3)}\, \frac{c_{\text{LOP}}^{(d=3)}}{c_{\text{LOP}}^{(d=2)}}\, \frac{c_\theta^{(d=2)}}{c_\theta^{(d=3)}} = 2.396 \tag{36}$$

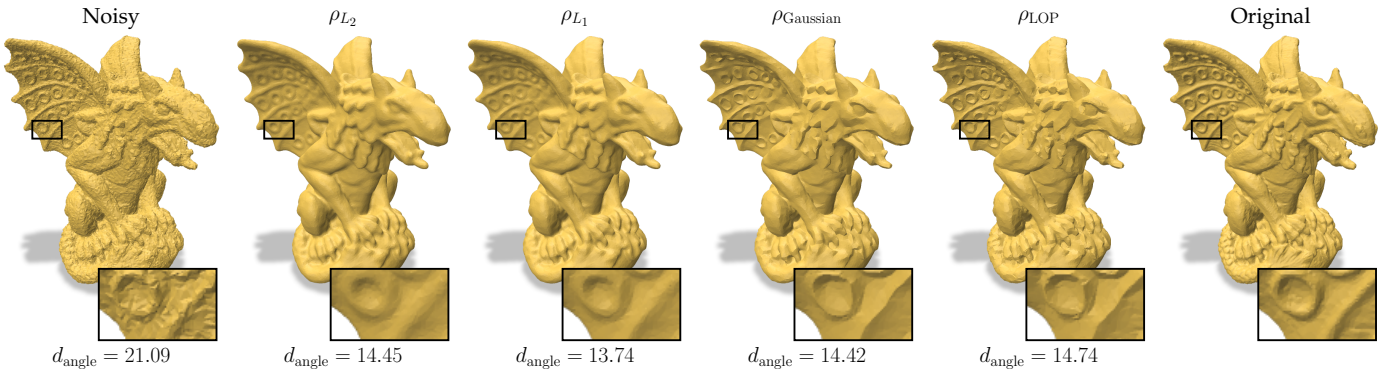


Fig. 10. Mesh denoising of the *Gargoyle* model (86 311 vertices, 172 610 faces) corrupted with $0.25\,\bar{l}_e$ uniform noise. Although the mean angular distance d_{angle} is slightly higher for the LOP loss ρ_{LOP}, features and finer details are better preserved. Unit: [°].

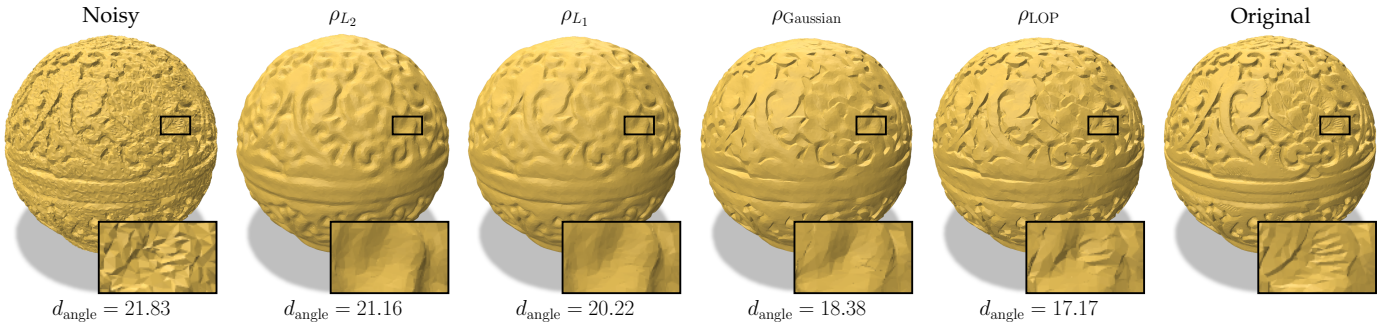


Fig. 11. Mesh denoising of the *Box* model (70 134 vertices, 140 259 faces) corrupted with $0.25\,\bar{l}_e$ uniform noise. Filtering with the LOP loss ρ_{LOP} results in the lowest mean angular distance d_{angle} and reconstructs fine details best. Unit: [°].

which consists of three terms: 1) the normalization constant of the kernel density estimate in (1); 2) the missing normalization constant of the kernel θ; and 3) a dimension-dependent correction factor. The last term models the different domains from which the density is accumulated as we consider a surface patch that corresponds to a 2D subspace embedded in the 3D space. Therefore, the integration domain of the density differs by one dimension which can be accounted for by the ratio of the normalization constants of both the actual density kernel K_{LOP} as well as the chosen kernel θ for density weight computation.

6.2 Evaluation of CLOP Kernel Approximation

We evaluated the approximation error of our fitted parameter set in Table 2 (see Section 5.2) against the original one proposed by CLOP [5]. First, we quantified systematic errors of the kernel approximation $\hat{K}_{\text{LOP}}$ by analyzing its standard deviation $\|\hat{\boldsymbol{\Sigma}}_{\text{LOP}}\|^{1/2}$ defined by the square root of the magnitude of its covariance matrix (26). Table 3 indicates that both CLOP and our approximation underestimate the actual value $\|\boldsymbol{\Sigma}_{\text{LOP}}\|^{1/2}$ and that the bias increases in higher dimensions. Although these errors are significantly lower for our approximation throughout all dimensions, they may still be noticeable. Our consistent approximation always overestimates the actual standard deviation and has a slightly higher error than the unconstrained variant in low dimensions up to $d = 5$. However, it becomes unbiased in the limit $d \to \infty$ and should be preferred in higher dimensions. We also considered minimizing the L_1 distance of the kernel approximation $\hat{K}_{\text{LOP}}$ (25) to the actual kernel K_{LOP} (11) by scaling the values $\hat{\sigma}_k$ with correction factors $1/b_{\text{opt}}$ to obtain an improved set of parameters which is shown in Fig. 8. Here, the error of our approximation is significantly lower than for CLOP both before and after optimal correction. Furthermore, the optimal scaling factors are similar to the ratios of the standard deviation for $d = 1$.

In addition to the theoretical analysis of the kernel approximations, we also measured the reconstruction error when replacing the actual kernel in the update step (7) of the projection with the respective approximation. For this purpose, we chose the *Block* model and uniformly sampled 50 000 target points $\mathcal{P}$ and 25 000 projection points $\mathcal{Q}$ respectively. We applied 100 iterations of WLOP as a smoothing operator with a large window size of $h = 25$ percent of the bounding box diagonal of $\mathcal{P}$ and a repulsion weight $\mu = 0.4$. Then, we measured the distance of each point to the (triangulated) surface of the reference point cloud as well as the mean point-surface distance

$$d_{\text{surface}}(\mathcal{X}, \mathcal{Y}) = \frac{1}{|\mathcal{X}|} \sum_i \min_j d(\boldsymbol{x}_i, t(y_j)) \tag{37}$$

where $t(y_j)$ denotes the j-th triangle of $\mathcal{Y}$. Fig. 9 shows the results of this point cloud smoothing operation. Whereas the CLOP approximation introduces higher errors at the edges of the sampled model due to the significantly underestimated standard deviation of the kernel, our approximation does not suffer from these artifacts.

6.3 Evaluation of Robust Loss Functions

We tested the LOP M-estimator with the corresponding loss ρ_{LOP} (32) against other popular choices, i.e. L_2, L_1, and Gaussian M-estimators, for normal filtering in the context of mesh denoising based on the iterative update step in (28) (see Section 5.3). For this, we used the *Gargoyle* (86 311 vertices, 172 610 faces) and *Box* (70 134 vertices, 140 259 faces) models and corrupted the vertices in random directions by $0.25\,\bar{l}_e$ uniform noise where $\bar{l}_e$ denotes the average face edge length. Then, we applied 50 iterations of normal filtering with $\sigma = 0.3$ for each face normal $\boldsymbol{n}$ within its geometric neighborhood of size $r = 1.5\,\bar{l}_e$, that is all normals whose face centers are traversable along the surface within a ball of size r. To avoid the singularity of the L_1 and LOP losses at $x = 0$, we only considered the neighboring face normals in the initial iteration and used all normals subsequently. For the second stage of the mesh denoising framework, we used the vertex update by Zhang *et al.* [83] with their default parameters of 20 iterations and $w = 0.001$ which avoids the triangle flipping problem. We evaluated the reconstruction error by the mean angular distance

$$d_{\text{angle}}(\mathcal{X}, \mathcal{Y}) = \frac{1}{|\mathcal{X}|} \sum_i \left[\frac{360}{2\pi} \arccos(\langle \boldsymbol{n}(x_i) | \boldsymbol{n}(y_i) \rangle) \right] \quad (38)$$

to the face normals of the ground truth mesh. Fig. 10 and 11 show comparisons between the L_2, L_1, Gaussian, and LOP loss. Whereas the L_2 loss leads to a very smooth surface, its L_1 counterpart is less sensitive to large normal variations within the local neighborhood and better preserves features. However, sharp edges cannot be reconstructed since all collected normals are considered in a global fashion. The Gaussian and LOP loss functions can be viewed as localized versions of the former losses and do not suffer from this limitation. Finer details being at a similar scale as the applied noise are hard to reconstruct and mostly smoothed out by all variants, but can be partially recovered by the LOP loss.

6.4 Evaluation of Neural Network Priors

We tested the application of different neural network priors for unsupervised point cloud denoising (see Section 5.4). For this purpose, we used the reference implementation of TotalDenoising [50] and trained the method in an unsupervised manner on the real-world large-scale *Paris-rue-Madame* dataset [84], which was captured by a LiDAR sensor and has no groundtruth data available. In particular, we employed the same training procedure and default parameter set in all cases, that is the scaling parameter $\alpha = 0.5$ as well as the standard deviation $\sigma = 0.5657$ for the kernel in the prior (34), and only replaced the Gaussian kernel by the LOP kernel. Finally, we used the trained models to denoise the test split of the dataset. Fig. 12 shows a comparison between the Gaussian and LOP kernel priors. While the LiDAR noise is reliably removed from the test datasets for both kernels, filtering with the Gaussian kernel leads to significant oversmoothing of smaller features and, in addition, to a noticable shrinkage of the point cloud. Furthermore, holes may be introduced at sharp features like edges since the Gaussian prior tends to move nearby points into clusters apart from these features. In contrast to this, the LOP kernel with equal standard deviation σ preserves higher frequency information and, in turn, sharp features without introducing holes. The shrinkage effect also is greatly mitigated which becomes apparent at thin structures, like the sash bars in the windows, where the width is significantly better preserved and closely matches the expected width from the input data.

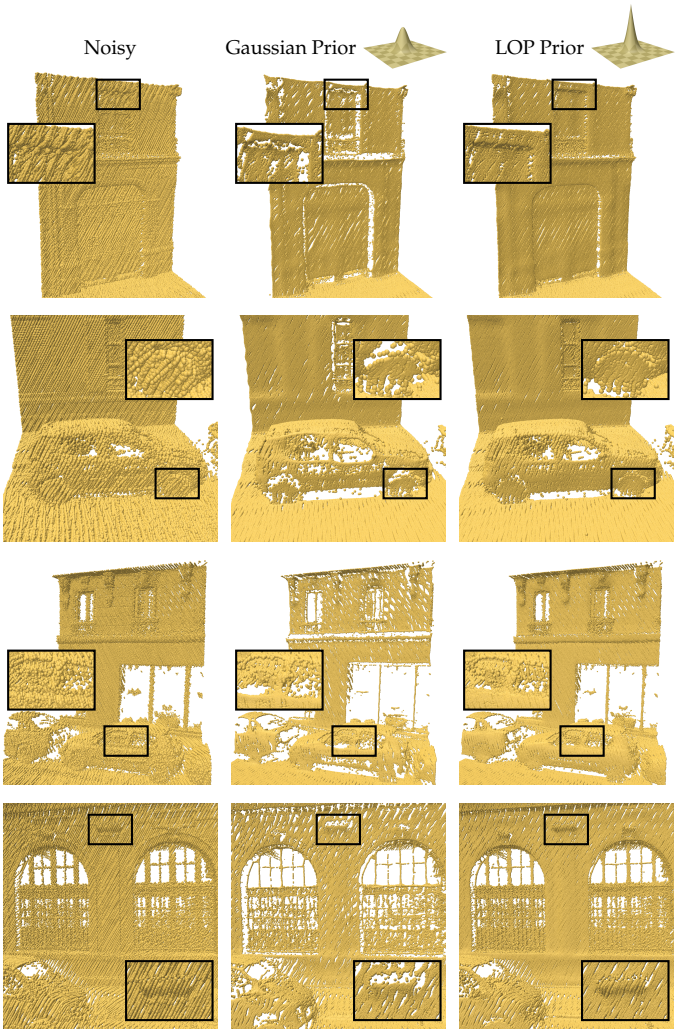


Fig. 12. Point cloud denoising of the *Paris-rue-Madame* dataset with TotalDenoising [50] for the Gaussian and LOP kernels as neural network priors. Performing the unsupervised training with the LOP prior better preserves features and edges and leads to significantly less oversmoothing and shrinkage. Furthermore, using the LOP prior also better handles the systematic pattern induced by the scanning process.

TABLE 4
Comparison Between Unsupervised Learning-based Point Cloud Denoising Methods. Unit: $[10^{-4}]$

	$\sigma_{\text{noise}} = 0.5$		$\sigma_{\text{noise}} = 1$		$\sigma_{\text{noise}} = 1.5$	
	d_{CD}	d_{P2M}	d_{CD}	d_{P2M}	d_{CD}	d_{P2M}
Score (Unsup.) [51]	**0.150**	**0.115**	0.404	0.315	0.821	0.609
TD (K_{Gaussian}) [50]	0.173	0.150	0.343	0.303	0.466	0.428
TD (K_{LOP})	0.175	0.148	**0.282**	**0.246**	**0.424**	**0.389**

In addition to these qualitative results, we also performed a quantitative comparison between recent learning-based approaches. For a fair comparison, we only considered unsupervised methods, that is TotalDenoising (TD) [50]

with the original Gaussian kernel prior as well as with our LOP kernel prior, and Score-based denoising [51] which, although being supervised, also supports training in an unsupervised manner. We used the *ModelNet40* dataset [85] provided by TD [50] that consists of 15 collected object classes with 5 training meshes and 2 test meshes per class, from which point clouds were sampled using Poisson disk sampling and corrupted by Gaussian noise with standard deviations of $\sigma_{\text{noise}} = 0.5, 1, 1.5$ percent of the bounding box diagonal. After training all methods with default parameters on this dataset, we applied them to denoise the test data using $1, 4, 7$ iterations with TD and $1, 1, 2$ iterations with Score-based denoising respectively for the different noise levels. We measured the performance with the commonly used Chamfer distance

$$d_{\text{CD}}(\mathcal{X}, \mathcal{Y}) = \frac{1}{|\mathcal{X}|} \sum_i d(\boldsymbol{x}_i, \mathcal{Y})^2 + \frac{1}{|\mathcal{Y}|} \sum_i d(\boldsymbol{y}_i, \mathcal{X})^2 \quad (39)$$

where $d(\boldsymbol{x}_i, \mathcal{Y}) = \min_j \|\boldsymbol{x}_i - \boldsymbol{y}_j\|$ denotes the nearest neighbor distance of $\boldsymbol{x}_i$ to the point cloud $\mathcal{Y}$ as well as with the point-to-mesh distance

$$d_{\text{P2M}}(\mathcal{X}, \mathcal{Y}) = \frac{1}{|\mathcal{X}|} \sum_i \min_j d(\boldsymbol{x}_i, t(y_j))^2 + \frac{1}{|\mathcal{Y}|} \sum_j \min_i d(\boldsymbol{x}_i, t(y_j))^2 \quad (40)$$

where $d(\boldsymbol{x}_i, t(y_j))$ is the distance of $\boldsymbol{x}_i$ to the triangle $t(y_j)$ as in (37). Table 4 shows the results of all methods averaged over the test data. At the low noise level $\sigma_{\text{noise}} = 0.5$, Score-based denoising achieves slightly better results than TotalDenoising where the performance is very similar between the Gaussian and the LOP kernel prior. On the other hand, applying the LOP kernel prior significantly increases the robustness of TotalDenoising at higher noise levels and also outperforms all other approaches with respect to both the Chamfer and the point-to-mesh distance.

7 Conclusions

We presented incomplete gamma kernels, a novel family of kernels generalizing LOP operators. By revisiting the classical localized L_1 estimator used in LOP, we revealed its relation to the Mean Shift framework via a novel kernel K_{LOP} and generalized this result to arbitrary localized L_p estimators. We derived several theoretical properties of the kernel family K_Γ concerning distributional, Mean Shift induced, and other aspects such as strict positive definiteness to obtain a deeper understanding of the operator's projection behavior. Furthermore, we illustrated several applications including an improved WLOP density weighting scheme, a more accurate kernel approximation for CLOP, incomplete gamma losses ρ_Γ as a novel set of robust loss functions, as well as better neural network priors and confirmed their effectiveness in a variety of quantitative and qualitative experiments. We expect that building upon the insights provided by our work will be beneficial for future developments on point cloud denoising as well as in many other related fields, including applications beyond pure denoising.

Acknowledgments

This work was supported by the DFG projects KL 1142/11-2 (DFG Research Unit FOR 2535 Anticipating Human Behavior) and NFDI4Culture (DFG Project Number 441958017). We thank the reviewers for their valuable comments which helped us to improve the quality of our manuscript.

References

[1] Y. Lipman, D. Cohen-Or, D. Levin, and H. Tal-Ezer, "Parameterization-free Projection for Geometry Reconstruction," *ACM Trans. Graph. (TOG)*, vol. 26, no. 3, 2007.
[2] H. Huang, D. Li, H. Zhang, U. Ascher, and D. Cohen-Or, "Consolidation of Unorganized Point Clouds for Surface Reconstruction," *ACM Trans. Graph. (TOG)*, vol. 28, no. 5, 2009.
[3] B. Liao, C. Xiao, L. Jin, and H. Fu, "Efficient feature-preserving local projection operator for geometry reconstruction," *Comput. Aided Des.*, vol. 45, no. 5, 2013.
[4] H. Huang, S. Wu, M. Gong, D. Cohen-Or, U. Ascher, and H. Zhang, "Edge-Aware Point Set Resampling," *ACM Trans. Graph. (TOG)*, vol. 32, no. 1, 2013.
[5] R. Preiner, O. Mattausch, M. Arikan, R. Pajarola, and M. Wimmer, "Continuous Projection for Fast L_1 Reconstruction," *ACM Trans. Graph. (TOG)*, vol. 33, no. 4, 2014.
[6] D. Zhang, X. Lu, H. Qin, and Y. He, "Pointfilter: Point Cloud Filtering via Encoder-Decoder Modeling," *IEEE Trans. Vis. Comput. Graph. (TVCG)*, vol. 27, no. 3, 2020.
[7] X. Xu, G. Geng, X. Cao, K. Li, and M. Zhou, "TDNet: transformer-based network for point cloud denoising," *Appl. Opt.*, vol. 61, no. 6, 2022.
[8] K. Fukunaga and L. Hostetler, "The estimation of the gradient of a density function, with applications in pattern recognition," *IEEE Trans. Inf. Theory*, vol. 21, no. 1, 1975.
[9] Y. Cheng, "Mean Shift, Mode Seeking, and Clustering," *IEEE Trans. Pattern Anal. Mach. Intell. (TPAMI)*, vol. 17, no. 8, 1995.
[10] D. Comaniciu and P. Meer, "Mean Shift: A Robust Approach toward Feature Space Analysis," *IEEE Trans. Pattern Anal. Mach. Intell. (TPAMI)*, vol. 24, no. 5, 2002.
[11] P. Madaan, A. Maiti, S. Anand, and S. Mittal, "Deep Mean Shift Clustering," Bachelor's Thesis, Indraprastha Institute of Information Technology, 2019.
[12] S. Arjomand Bigdeli and M. Zwicker, "Image Restoration using Autoencoding Priors," in *Proc. Int. Joint Conf. Comput. Vis., Imaging Comput. Graph. Theory Appl.*, vol. 5, 2018.
[13] G. Alain and Y. Bengio, "What Regularized Auto-Encoders Learn from the Data-Generating Distribution," *J. Mach. Learn. Res.*, vol. 15, no. 1, 2014.
[14] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle, "Surface Reconstruction from Unorganized Points," in *Proc. Annu. Conf. Comput. Graph. Interact. Techn. (SIGGRAPH)*, 1992.
[15] J. C. Carr *et al.*, "Reconstruction and Representation of 3D Objects with Radial Basis Functions," in *Proc. Annu. Conf. Comput. Graph. Interact. Techn. (SIGGRAPH)*, 2001.
[16] M. Alexa, J. Behr, D. Cohen-Or, S. Fleishman, D. Levin, and C. T. Silva, "Computing and Rendering Point Set Surfaces," *IEEE Trans. Vis. Comput. Graph. (TVCG)*, vol. 9, no. 1, 2003.
[17] N. Amenta and Y. J. Kil, "Defining Point-Set Surfaces," *ACM Trans. Graph. (TOG)*, vol. 23, no. 3, 2004.
[18] S. Fleishman, D. Cohen-Or, and C. T. Silva, "Robust Moving Least-squares Fitting with Sharp Features," *ACM Trans. Graph. (TOG)*, vol. 24, no. 3, 2005.
[19] A. C. Öztireli, G. Guennebaud, and M. Gross, "Feature Preserving Point Set Surfaces based on Non-Linear Kernel Regression," in *Comput. Graph. Forum (CGF)*, vol. 28, no. 2, 2009.
[20] E. A. L. Narváez and N. E. L. Narváez, "Point cloud denoising using robust principal component analysis," in *Int. Conf. Comput. Graph. Theory Appl.*, 2006.
[21] E. Mattei and A. Castrodad, "Point Cloud Denoising via Moving RPCA," in *Comput. Graph. Forum (CGF)*, vol. 36, no. 8, 2017.
[22] X. Lu, S. Wu, H. Chen, S.-K. Yeung, W. Chen, and M. Zwicker, "GPF: GMM-inspired Feature-preserving Point Set Filtering," *IEEE Trans. Vis. Comput. Graph. (TVCG)*, vol. 24, no. 8, 2017.
[23] Y. Sun, S. Schaefer, and W. Wang, "Denoising point sets via L_0 minimization," *Comput. Aided Geom. Des.*, vol. 35, 2015.

[24] X. Cheng, M. Zeng, J. Lin, Z. Wu, and X. Liu, "Efficient L_0 resampling of point sets," *Comput. Aided Geom. Des.*, vol. 75, 2019.

[25] H. Avron, A. Sharf, C. Greif, and D. Cohen-Or, "ℓ_1-Sparse Reconstruction of Sharp Point Set Surfaces," *ACM Trans. Graph. (TOG)*, vol. 29, no. 5, 2010.

[26] E. Leal, G. Sanchez-Torres, and J. W. Branch, "Sparse Regularization-Based Approach for Point Cloud Denoising and Sharp Features Enhancement," *Sensors*, vol. 20, no. 11, 2020.

[27] J. Digne, S. Valette, and R. Chaine, "Sparse Geometric Representation Through Local Shape Probing," *IEEE Trans. Vis. Comput. Graph. (TVCG)*, vol. 24, no. 7, 2017.

[28] J. Zeng, G. Cheung, M. Ng, J. Pang, and C. Yang, "3D Point Cloud Denoising Using Graph Laplacian Regularization of a Low Dimensional Manifold Model," *IEEE Trans. Image Process.*, vol. 29, 2019.

[29] W. Hu, X. Gao, G. Cheung, and Z. Guo, "Feature Graph Learning for 3D Point Cloud Denoising," *IEEE Trans. Image Process.*, vol. 68, 2020.

[30] C. Dinesh, G. Cheung, and I. V. Bajić, "Point Cloud Denoising via Feature Graph Laplacian Regularization," *IEEE Trans. Image Process.*, vol. 29, 2020.

[31] M. A. Irfan and E. Magli, "Joint Geometry and Color Point Cloud Denoising Based on Graph Wavelets," *IEEE Access*, vol. 9, 2021.

[32] Y. Zheng, G. Li, S. Wu, Y. Liu, and Y. Gao, "Guided point cloud denoising via sharp feature skeletons," *Vis. Comput.*, vol. 33, no. 6, 2017.

[33] C. Shi *et al.*, "Three-dimensional point cloud denoising via a gravitational feature function," *Appl. Opt.*, vol. 61, no. 6, 2022.

[34] W. Hu, Q. Hu, Z. Wang, and X. Gao, "Dynamic Point Cloud Denoising via Manifold-to-Manifold Distance," *IEEE Trans. Image Process.*, vol. 30, 2021.

[35] O. Remil, Q. Xie, X. Xie, K. Xu, and J. Wang, "Surface Reconstruction with Data-driven Exemplar Priors," *Comput. Aided Des.*, vol. 88, 2017.

[36] G. Rosman, A. Dubrovina, and R. Kimmel, "Patch-Collaborative Spectral Point-Cloud Denoising," in *Comput. Graph. Forum (CGF)*, vol. 32, no. 8, 2013.

[37] X. Lu, S. Schaefer, J. Luo, L. Ma, and Y. He, "Low Rank Matrix Approximation for 3D Geometry Filtering," *IEEE Trans. Vis. Comput. Graph. (TVCG)*, vol. 28, no. 4, 2020.

[38] H. Chen, M. Wei, Y. Sun, X. Xie, and J. Wang, "Multi-Patch Collaborative Point Cloud Denoising via Low-Rank Recovery with Graph Constraint," *IEEE Trans. Vis. Comput. Graph. (TVCG)*, vol. 26, no. 11, 2019.

[39] D. Zhu *et al.*, "Non-local Low-rank Point Cloud Denoising for 3D Measurement Surfaces," *IEEE Trans. Instrum. Meas.*, 2022.

[40] F. Zaman, Y. P. Wong, and B. Y. Ng, "Density-Based Denoising of Point Cloud," in *Int. Conf. Robot., Vis., Signal Process. Power Appl.*, 2017.

[41] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2017.

[42] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space," *Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 30, 2017.

[43] W. Yifan, S. Wu, H. Huang, D. Cohen-Or, and O. Sorkine-Hornung, "Patch-based Progressive 3D Point Set Upsampling," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2019.

[44] J. Choe, B. Joung, F. Rameau, J. Park, and I. S. Kweon, "Deep Point Cloud Reconstruction," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2022.

[45] C. Duan, S. Chen, and J. Kovacevic, "3D Point Cloud Denoising via Deep Neural Network based Local Surface Estimation," in *IEEE Int. Conf. Acoust. Speech Signal Process.*, 2019.

[46] M.-J. Rakotosaona, V. La Barbera, P. Guerrero, N. J. Mitra, and M. Ovsjanikov, "PointCleanNet: Learning to Denoise and Remove Outliers from Dense Point Clouds," in *Comput. Graph. Forum (CGF)*, vol. 39, no. 1, 2020.

[47] L. Yu, X. Li, C.-W. Fu, D. Cohen-Or, and P.-A. Heng, "EC-Net: an Edge-aware Point set Consolidation Network," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018.

[48] H. Zhou, K. Chen, W. Zhang, H. Fang, W. Zhou, and N. Yu, "DUP-Net: Denoiser and Upsampler Network for 3D Adversarial Point Clouds Defense," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, 2019.

[49] F. Pistilli, G. Fracastoro, D. Valsesia, and E. Magli, "Learning Graph-Convolutional Representations for Point Cloud Denoising," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2020.

[50] P. Hermosilla, T. Ritschel, and T. Ropinski, "Total Denoising: Unsupervised Learning of 3D Point Cloud Cleaning," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, 2019.

[51] S. Luo and W. Hu, "Score-Based Point Cloud Denoising," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, 2021.

[52] H. Chen, B. Du, S. Luo, and W. Hu, "Deep Point Set Resampling via Gradient Fields," *arXiv preprint arXiv:2111.02045*, 2021.

[53] Y. Zhao, H. Zheng, Z. Wang, J. Luo, and E. Y. Lam, "Point Cloud Denoising via Momentum Ascent in Gradient Fields," *arXiv preprint arXiv:2202.10094*, 2022.

[54] R. Roveri, A. C. Öztireli, I. Pandele, and M. Gross, "PointProNets: Consolidation of Point Clouds with Convolutional Neural Networks," in *Comput. Graph. Forum (CGF)*, vol. 37, no. 2, 2018.

[55] W. Yifan, F. Serena, S. Wu, C. Öztireli, and O. Sorkine-Hornung, "Differentiable Surface Splatting for Point-based Geometry Processing," *ACM Trans. Graph. (TOG)*, vol. 38, no. 6, 2019.

[56] S. Luo and W. Hu, "Differentiable Manifold Reconstruction for Point Cloud Denoising," in *Proc. ACM Int. Conf. Multimedia*, 2020.

[57] R. Hanocka, G. Metzer, R. Giryes, and D. Cohen-Or, "Point2Mesh: A Self-Prior for Deformable Meshes," *ACM Trans. Graph. (TOG)*, vol. 39, no. 4, 2020.

[58] C. Huang, R. Li, X. Li, and C.-W. Fu, "Non-Local Part-Aware Point Cloud Denoising," *arXiv preprint arXiv:2003.06631*, 2020.

[59] Y. Chen *et al.*, "Shape Self-Correction for Unsupervised Point Cloud Understanding," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, 2021.

[60] A. Mao, Z. Du, Y.-H. Wen, J. Xuan, and Y.-J. Liu, "PD-Flow: A Point Cloud Denoising Framework with Normalizing Flows," *arXiv preprint arXiv:2203.05940*, 2022.

[61] H. Chen, Z. Wei, X. Li, Y. Xu, M. Wei, and J. Wang, "RePCD-Net: Feature-Aware Recurrent Point Cloud Denoising Network," *Int. J. Comput. Vis. (IJCV)*, 2022.

[62] C. Grillenzoni, "Design of Blurring Mean-Shift Algorithms for Data Classification," *J. Classif.*, vol. 33, no. 2, 2016.

[63] G. Beck, T. Duong, M. Lebbah, H. Azzag, and C. Cérin, "A Distributed and Approximated Nearest Neighbors Algorithm for an Efficient Large Scale Mean Shift Clustering," *J. Parallel Distrib. Comput.*, vol. 134, 2019.

[64] J. Jang and H. Jiang, "MeanShift++: Extremely Fast Mode-Seeking With Applications to Segmentation and Object Tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2021.

[65] S. Arjomand Bigdeli, M. Zwicker, P. Favaro, and M. Jin, "Deep Mean-Shift Priors for Image Restoration," *Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 30, 2017.

[66] X. Li, Z. Hu, and F. Wu, "A note on the convergence of the mean shift," *Pattern Recognit.*, vol. 40, no. 6, 2007.

[67] T.-L. Chen, "On the convergence and consistency of the blurring mean-shift process," *Ann. Inst. Stat. Math.*, vol. 67, no. 1, 2015.

[68] Y. A. Ghassabeh, "A sufficient condition for the convergence of the mean shift algorithm with Gaussian kernel," *J. Multivar. Anal.*, vol. 135, 2015.

[69] K. Huang, X. Fu, and N. Sidiropoulos, "On Convergence of Epanechnikov Mean Shift," in *Proc. AAAI Conf. Artif. Intell.*, vol. 32, no. 1, 2018.

[70] R. Yamasaki and T. Tanaka, "Properties of Mean Shift," *IEEE Trans. Pattern Anal. Mach. Intell. (TPAMI)*, vol. 42, no. 9, 2019.

[71] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol, "Extracting and Composing Robust Features with Denoising Autoencoders," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2008.

[72] E. Weiszfeld, "Sur le point pour lequel la somme des distances de n points donnés est minimum," *Tohoku Math. J.*, vol. 43, 1937.

[73] H. Bateman, *Higher Transcendental Functions*. McGraw-Hill Book Company, 1953, vol. 2.

[74] E. M. Stein and G. Weiss, *Introduction to Fourier Analysis on Euclidean Spaces*. Princeton University Press, 1971.

[75] P. Natalini and B. Palumbo, "Inequalities for the incomplete gamma function," *Math. Inequal. Appl.*, vol. 3, no. 1, 2000.

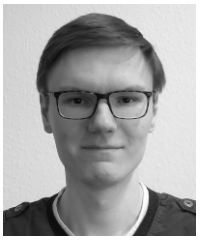
[76] S. Faigenbaum-Golovin and D. Levin, "Manifold reconstruction and denoising from scattered data in high dimension," *J. Comput. Appl. Math.*, vol. 421, 2023.

[77] R. L. Schilling, R. Song, and Z. Vondracek, *Bernstein Functions: Theory and Applications*. De Gruyter, 2012.

[78] I. J. Schoenberg, "Metric Spaces and Completely Monotone Functions," *Ann. Math.*, 1938.

[79] F. R. Hampel, E. M. Ronchetti, P. J. Rousseeuw, and W. A. Stahel, *Robust Statistics: The Approach Based on Influence Functions*. Wiley, 1986.

[80] M. J. Black, G. Sapiro, D. H. Marimont, and D. Heeger, "Robust Anisotropic Diffusion," *IEEE Trans. Image Process.*, vol. 7, no. 3, 1998.
[81] S. K. Yadav, M. Skrodzki, E. Zimmermann, and K. Polthier, "Surface Denoising based on Normal Filtering in a Robust Statistics Framework," in *Proc. Forum "Math-for-Industry" 2018*, 2021.
[82] J. T. Barron, "A General and Adaptive Robust Loss Function," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2019.
[83] J. Zhang, B. Deng, Y. Hong, Y. Peng, W. Qin, and L. Liu, "Static/Dynamic Filtering for Mesh Geometry," *IEEE Trans. Vis. Comput. Graph. (TVCG)*, vol. 25, no. 4, 2018.
[84] A. Serna, B. Marcotegui, F. Goulette, and J.-E. Deschaud, "Paris-rue-Madame database: a 3D mobile laser scanner dataset for benchmarking urban detection, segmentation and classification methods," in *Int. Conf. Pattern Recognit., Appl. Methods*, 2014.
[85] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, and J. Xiao, "3D ShapeNets: A Deep Representation for Volumetric Shapes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2015.



Patrick Stotko is a Ph.D. Student with the Visual Computing department at the University of Bonn, Germany. He received the B.Sc. and M.Sc. degrees in Computer Science from the University of Bonn, Germany, in 2014 and 2016 respectively. His research interests include interactive geometry and appearance reconstruction techniques for AR and VR applications.

Michael Weinmann is an Assistant Professor with the Intelligent Systems department at Delft University of Technology, The Netherlands. He received the Dipl.-Ing. degree in Electrical Engineering and Information Technology from the University of Karlsruhe, Germany, in 2009 and the Ph.D. degree in Computer Science from the University of Bonn, Germany, in 2016. His current research interests include object and scene digitization, digital material appearance, scene representations, scene interpretation, data visualization, live telepresence, machine learning, computational arts and cultural heritage.

Reinhard Klein is a Professor with the University of Bonn, Germany, and the Director of the Visual Computing department. He received the Dipl.-Math. degree in Mathematics and the Ph.D. degree in Computer Science from the University of Tübingen, Germany, in 1989 and 1995 respectively. In 1999, he became Associate Professor at the University of Darmstadt, Germany, and Head of the research group Animation and Image Communication at the Fraunhofer Institute for Computer Graphics. Since 2000, he is a Professor at the University of Bonn, Germany, and Director of the Institute of Computer Science II – Visual Computing. His research interests include computer graphics, material appearance, rendering, geometry processing and architectural geometry.