A Large-Scale Evaluation Of Shape-Aware Neighborhood Weights And Neighborhood Sizes☆

Martin Skrodzki [a,b,c,]*, Eric Zimmermann [d]

[a] *ICERM, Brown University, Providence, RI, USA*
[b] *RIKEN iTHEMS, Wako, Saitama, Japan*
[c] *CGV, TU Delft, Delft, The Netherlands*
[d] *Institute of Mathematics, Freie Universität Berlin, Berlin, Germany*

ARTICLE INFO

ABSTRACT

In this paper, we define and evaluate a weighting scheme for neighborhoods in point sets. Our weighting takes the shape of the geometry, i.e., the normal information, into account. This causes the obtained neighborhoods to be more reliable in the sense that connectivity also depends on the orientation of the point set. We utilize a sigmoid to define the weights based on the normal variation. For an evaluation of the weighting scheme, we turn to a Shannon entropy model for feature classification that can be proven to be non-degenerate for our family of weights. Based on this model, we evaluate our weighting terms on a large scale of both clean and real-world models. This evaluation provides results regarding the choice of optimal parameters within our weighting scheme. Furthermore, the large-scale evaluation also reveals that neighborhood sizes should not be fixed globally when processing models. Finally, we highlight the applicability of our weighting scheme within the application context of denoising.

1. Introduction

Point sets arise naturally in many kinds of 3D acquisition processes, like, e.g., 3D laser-scanning. As early as 1985, they have been recognized as fundamental shape representations in computer graphics, see [1]. Ever since, they have been used in diverse applications, e.g., in archaeology [2], face recognition [3], or traffic accident analysis [4].

Despite their versatility and their advantages—like easy acquisition and low storage costs—point sets have a significant downside to them when compared with mesh representations: They are not equipped with connectivity information. This is mostly due to the acquisition process. Consider for example a manually guided scanning device. The operator will scan those areas of the real-world objects with very sharp features multiple times. Consequently, occlusion is prevented and the whole geometry is captured. Even though each scan can provide connectivity information on the respectively acquired points, the complete point set obtained via registration of the individual scans (see, e.g., [5]) does not provide global connectivity information in general. Thus, a notion of neighborhoods has to be defined and computed for each point.

Many definitions of neighborhoods, combinatorial or geometric, with global or local parameters, have been proposed and discussed (see Section 2). Furthermore, the concept of weighting neighboring points is not new. For example, the pure selection of a neighborhood causes an equal treatment of all neighbors. Aside from this, isotropic weighting is one common way, evaluating Euclidean distances via a Gaussian weighting function. This provides closer points with higher influence (see, e.g., [6]). Additionally, other point set information can be incorporated, like density or distribution (see, e.g., [7] or [8]). The inclusion of normal deviation in the area of anisotropic weighting has also been considered and discussed before (see [9,10]).

The research work presented here aims at investigating anisotropic weighting terms in a broad framework, which includes usual weighting choices such as equal weights or sharp

Martin Skrodzki (M. Skrodzki).
E-mail addresses: mail@ms-math-computer.science (M. Skrodzki), eric.zimmermann@fu-berlin.de (E. Zimmermann).
URLs: https://ms-math-computer.science (M. Skrodzki), https://userpage.fu-berlin.de/ezimmermann/ (E. Zimmermann).

cut-off weights[1] (Section 3). Our evaluation is processed via a Shannon entropy model (Section 4), which is based on the work of [11,12]. Furthermore, we aim at evaluating the weighting scheme on a large scale. This is to prevent over-interpretation of findings obtained from a very small set of models. Overall, the contributions of this work are:

- Definition of a shape-aware neighborhood weighting utilizing sigmoid function weights based on normal variation;
- Presentation of a Shannon entropy evaluation model that can be proven to be non-degenerate on our inputs;
- Large scale experimental evaluation of the proposed neighborhood weighting concept;
- Discussion of the results with respect to both neighborhood weighting and neighborhood sizes.

While the content of this paper is deeply routed in the field of traditional computer-aided design, in our concluding Section 6, we will provide an outlook and several thoughts on the application of the presented techniques within the context of machine learning.

2. Related work

Neighborhoods are very important in point set processing, as almost all algorithmic approaches rely on them. A common choice is to use heuristics to determine sufficient notions like the size of a combinatorial or metric neighborhood. In the following, we recall works discussing heuristic neighborhood definitions. Several works have advanced from simple heuristics and derive more involved notions for better fitting neighborhood definitions in different contexts. These are mainly obtained from error functionals, which we will also discuss.

2.1. Heuristics

Most works consider either a combinatorial k-nearest neighborhood $\mathcal{N}_k(\cdot)$ or a metric ball $B_r(\cdot)$ inducing a neighborhood. Both of these notions have parameters to be tuned, namely the number of neighbors k or the radius r of the neighborhood. Several works have been presented introducing heuristics to find appropriate values for k or r in different scenarios. The authors of [6] for instance use a global radius and change it to affect the running time of their algorithm. In [13], the authors fix a combinatorial number k of neighbors to be sought. Then, for each point p_i from the considered point set P, these k neighbors are found, which fixes a radius r_i to the farthest of them. Finally, the neighbors within radius $r_i/3$ are used. Therefore, their approach resembles the geometric neighborhood in a local manner.

The method used in [14] is more involved. The authors recognize that both a too large or too small radius r lead to problems and thus aim for a local adaption like [13]. A local density estimate δ_i around each point $p_i \in P$ is computed from the smallest ball centered at p_i, containing $\mathcal{N}_k(p_i)$, where k is found experimentally to be best chosen from $\{6, \ldots, 20\} \subset \mathbb{N}$. Given the radius r_i of this ball, the local density is set to be $\delta_i = k/r_i^2$. In a second step, a smooth density function δ is interpolated from the local density estimates δ_i, hence this weighting involves the incorporation of density-information into the weight assignment.

In the context of surface reconstruction, the authors of [15] discuss several choices for neighborhoods and corresponding weights. While two of the three presented methods simply use geometric neighborhoods, the third method takes a different approach. Namely, the authors collect all neighbors of p_i in a

"large" ball ([15, page 7]) around p_i. Then, they fit a plane to this preliminary neighborhood and project all neighbors and p_i onto this plane. On the projections, a Delaunay triangulation is built and the induced neighborhood of the triangulation is used in the following computations, which localizes their approach and respects different point distributions.

A completely different route is taken by [16]. The authors first calculate features of a point set based on differently sized neighborhoods. Then, they use a training procedure to find the combination of neighborhood sizes that provides the best separation of different feature classes.

The inclusion of normal deviation and hence anisotropic weighting into neighborhood concepts is part of the work [9]. The approach of the authors is to use a weighted principal component analysis, which fits our evaluation model. However, they rely on a global neighborhood size and assign sharp cut-off weights while we allow for changing neighborhood sizes and smooth weighting terms.

2.2. Error functionals

While the approaches presented above are based on heuristics, some works try to deduce an optimal k for the k nearest neighborhoods based on error functions. For instance, the authors of [17] work in the context of the Moving Least Squares (MLS) framework (see [6,18–20]) for function approximation. The authors perform an extensive error analysis to quantify the approximation error both independent and depending on the given data. Finally, they obtain an error functional. This is then evaluated for different neighborhood sizes k. The neighborhood $\mathcal{N}_k$ yielding the smallest error is finally chosen to be used in the actual MLS approximation.

In contrast, the authors of [21] deduce an error bound on the normal estimation obtained from different neighborhood sizes. Utilizing this error functional, they obtain the best suited neighborhood size for normal computation. The work of [17] heavily depends on the MLS framework in which the error analysis is deduced, while the work of [21] depends on the framework of normal computation.

The authors of [12] take a more general approach in the context of segmentation of 3D point sets. They also use the concept of combinatorial neighborhoods, going back to results of [11,22]. In order to choose an optimal value for k, the authors turn to the covariance matrix, which is symmetric and positive-semi-definite. Thus, the matrix has three non-negative eigenvalues. Following an idea of [23], in the work of [14], the authors grow a neighborhood and consider a surface variation as a measure to grow a neighborhood around each point p_i. The same quantity is used by [24]. However, the authors of [14] do not grow a neighborhood, but choose a size k for it according to a consistent curvature level. The authors of [12] do not stop at these information, but proceed to consider three more quantities derived from the eigenvalues of the covariance matrix reflecting point set features, see [11,12]. Afterwards, following the concept of entropy by Shannon [25], they evaluate combinatorial and geometric neighborhood sizes via two error measures (see Section 4 for a detailed discussion).

3. Sigmoid weights

In contrast to the works listed above, our approach aims at integrating the shape of the geometry, i.e., the normal information, into the neighborhood definition. We will do so by enriching a given combinatorial neighborhood with a set of weights that are dependent on the normal variation within the neighborhood.

---

[1] We consider the case of cut-off weights if starting from a given deviation, all points with greater or equal deviation are attributed weight 0.

To ensure a smooth transition of weights, we apply a sigmoid function to the angle deviation of the normals.

Given a set of points $P = \{p_i \mid i \in [n]\}$, $n \in \mathbb{N}$, corresponding oriented unit-length normals $n_i \in \mathbb{S}^2$, and local neighborhoods $\mathcal{N}_i \subset [n]$ for every $i \in [n]$. For a given weighting function

$$\phi : [0, \pi] \to [0, 1], \tag{1}$$

we obtain the following weights

$$w_{ij} = \phi\left(\angle(n_i, n_j)\right) \quad \text{for } i \in [n], \ j \in \mathcal{N}_i. \tag{2}$$

The argument of ϕ is the deviation of the normals measured by their angle, which ranges from 0 to π. We turn to this formulation, because it has an obvious geometric interpretation. In order to have an efficient implementation of the presented techniques, the scenario can be reformulated in terms of the scalar product of the normals, which avoids the costly computation of arccos.

Note that by the symmetry of the angle, the weights are symmetric, i.e., $w_{ij} = w_{ji}$. The weighting function ϕ shall assign non-negative weights between 0 and 1. These weights should correspond to the similarity of the corresponding normals, i.e., a small angle should result in weights close to or equal 1, while a large angle should yield weights close to or equal 0.

Our choice for the weighting function is a sigmoid. A sigmoid function is visually characterized by its shape of an "S"-curve, even though it is mirrored in our scenario, see Fig. 1. We will consider a family of sigmoid functions that provide different interpolations between 1 and 0. The family is based on the trigonometric cosine function. It is related to the sigmoid used in [26], however, we alter it to be a monotonic falling curve between 1 and 0 on the interval $[0, \pi]$.

Definition 1 (*Cosine-Sigmoid*). Consider two given thresholds $a \in [0, \pi]$ and $b \in [a, \pi]$. Then, we define the sigmoid weighting function $\mathrm{sig}_{a,b}^{\cos} : [0, \pi] \to [0, 1]$ as

$$x \mapsto \begin{cases} 1 & x \in [0, a[\,, \\ \frac{1}{2}\cos\left(\frac{\pi(x-a)}{b-a}\right) + \frac{1}{2} & x \in [a, b]\,, \\ 0 & x \in \,]b, \pi]\,. \end{cases} \tag{3}$$

Note that for $a \neq b$ this function is $\mathcal{C}^1$ and smoothly transitions from 1 to 0. In particular, both boundary values are included, i.e., points can be given both weights 1 and 0, which corresponds to fully taking them into account or to not taking them into account at all. The threshold parameter $a \in [0, \pi]$ translates the curve along the x-axis and controls where the cosine curve starts. Similarly, the threshold parameter $b \in [a, \pi]$ controls where the cosine curve ends, i.e., the curve's decline is controlled by the distance between these two thresholds. In particular, when choosing $a = b = \pi$, all inputs obtain uniform weight 1 while for $a = b$, the function models a sharp cut-off at the chosen threshold. This allows us to relate our weights to the uniform weights used in [12] and to the sharp cut-off weights of [9], respectively.

4. Evaluation model

Having presented the set of neighborhood weights in Eq. (2) and the corresponding weighting function in Eq. (3) in the previous section, we will now describe the mathematical background of our evaluation process. For this, we turn to the information measures originally introduced by Shannon [25]. Specifically, we will use a variation of the quantities derived in [11,12] as we will present in Section 4.1. First, we will establish the necessary notation and preliminary results.

Consider the covariance matrices $C_i \in \mathbb{R}^{3 \times 3}$ given by

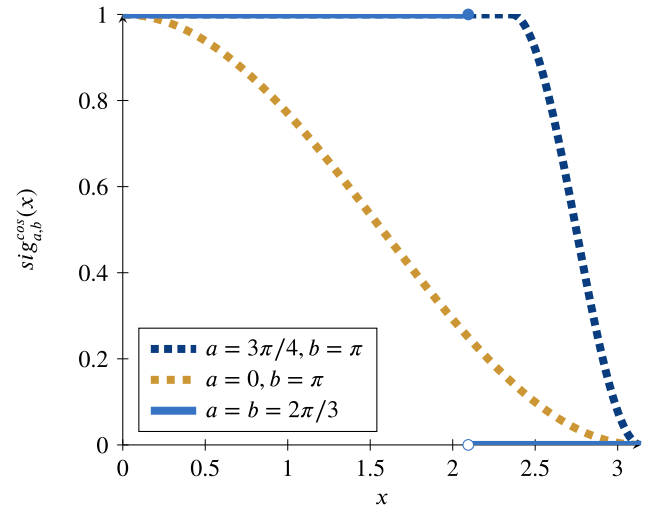$$C_i := \sum_{j \in \mathcal{N}_i} w_{ij}(p_j - \bar{p}_i)(p_j - \bar{p}_i)^T, \tag{4}$$

Fig. 1. Plots of the sigmoid $\mathrm{sig}_{a,b}^{\cos}(x)$ for three parameter choices on the domain $[0, \pi]$.

with $i \in [n]$, where $\bar{p}_i = \frac{1}{|\mathcal{N}_i|} \sum_{j \in \mathcal{N}_i} p_j$ is the barycenter of the neighborhood of p_i, thus $(p_i - \bar{p}_i)$ is a column vector in $\mathbb{R}^3$ and $(p_j - \bar{p}_i)^T \in \mathbb{R}^{1 \times 3}$ its transpose, i.e., a row-vector. The weights w_{ij} are chosen according to Eq. (2). The covariance matrix C_i is symmetric and positive-semi-definite. Thus, it has three non-negative eigenvalues, which in the following we will denote by

$$\lambda_i^1 \geq \lambda_i^2 \geq \lambda_i^3 \geq 0. \tag{5}$$

Depending on the neighborhood $\mathcal{N}_i$ and the assigned weights w_{ij}, we can prove the following proposition about the covariance matrix C_i.

Proposition 1 (*Non-degenerate Covariance Matrix*). *For a set of points $P = \{p_i \mid i \in [n]\}$, fix a point $p_i \in P$ and its neighborhood $\mathcal{N}_i \subseteq [n]$, and consider the function $\mathrm{sig}_{a,b}^{\cos}$ from Eq. (3) as well as the covariance matrix C_i given in Eq. (4). Assume there are $\ell_1, \ell_2 \in \mathcal{N}_i$, $\ell_1 \neq \ell_2$ such that $p_{\ell_1} \neq p_{\ell_2}$ and $n_{\ell_1} \neq -n_{\ell_2}$. Then, for some $b \in [a, \pi]$, the sum of all eigenvalues of C_i is strictly positive, independent of the choice of $a \in [0, \pi]$.*

Note, that a non-degenerate covariance matrix can trivially be obtained by setting $a = \pi$. However, the proposition makes an even stronger statement, namely that degeneracy can be obtained independent from the choice of a. Its proof follows from the observation that the weights w_{ij} are non-negative, as are all eigenvalues of C_i since C_i is positive semi-definite. Thus, the sum of the eigenvalues is 0 if and only if all eigenvalues are. By a case distinction on the zero set of the function $\mathrm{sig}_{a,b}^{\cos}$, we can then prove that there exists some $b \in [a, \pi]$ which results in strictly positive weights, which proves the proposition.

4.1. Non-degenerate covariance matrix

Given the assumptions of Proposition 1, we can assume that $C_i \neq 0 \in \mathbb{R}^{3 \times 3}$. Therefore, we can derive certain quantities from the eigenvalues of the covariance matrix. In our context, we will consider the linearity L_λ, planarity P_λ, and scattering S_λ. These are given by

$$L_i^\lambda = \frac{\lambda_i^1 - \lambda_i^2}{\lambda_i^1}, \quad P_i^\lambda = \frac{\lambda_i^2 - \lambda_i^3}{\lambda_i^1}, \quad S_i^\lambda = \frac{\lambda_i^3}{\lambda_i^1} \tag{6}$$

and represent 1D, 2D, and 3D features in the point set, respectively. See [11] for a derivation and a detailed explanation of these
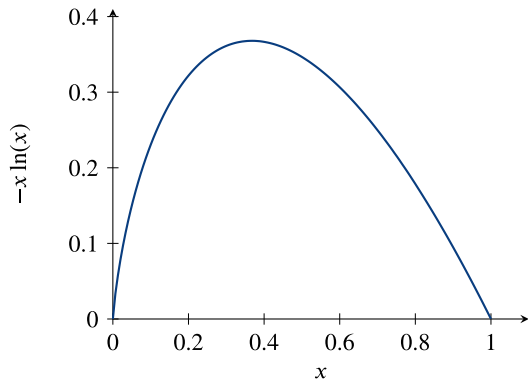
Fig. 2. Plot of the summand $-x\ln(x)$ from Eq. (7) for $x \in [0, 1]$ as all arguments L_i^λ, P_i^λ, and S_i^λ are taken from [0, 1].

quantities. As $C_i \neq 0$, we have $\lambda_i^1 \neq 0$, therefore the quantities in Eq. (6) are well-defined. Furthermore, because of the ordering of the eigenvalues given in Eq. (5), we have $L_i^\lambda, P_i^\lambda, S_i^\lambda \in [0, 1]$. Hence, as

$$L_i^\lambda + P_i^\lambda + S_i^\lambda = 1,$$

each of these three quantities can be interpreted as the probability of the considered point to be part of an intrinsic 1D, 2D, or 3D part of the geometry. The authors of [11,12] consider the error

$$E_i^{\dim} = -L_i^\lambda \ln(L_i^\lambda) - P_i^\lambda \ln(P_i^\lambda) - S_i^\lambda \ln(S_i^\lambda). \tag{7}$$

See Fig. 2 for a plot of each summand of the equation. Note that while $\lim_{x\to 0} \ln(x) = \infty$ it is $\lim_{x\to 0} x\ln(x) = 0$, which follows from rewriting it as quotient and applying L'Hôpital's rule. Practically, the error measure $E_i^{\dim}$ assesses to what extent the neighborhood $\mathcal{N}_i$ indicates a corner, an edge point, or a planar point of the geometry. In particular, the extreme cases

$$(\lambda_i^1, \lambda_i^2, \lambda_i^3) \in \{(\rho, 0, 0), (\rho, \rho, 0), (\rho, \rho, \rho) \mid \rho \in \mathbb{R}_{>0}\} \tag{8}$$

all obtain $E_i^{\dim} = 0$. That is to say that if a point p_i can be clearly classified as part of a linear, planar, or scattered segment of the point cloud, the classification error $E_i^{\dim}$ will indicate this.

Note that in general applications, these extreme cases are unlikely to occur. In particular in the presence of noise, the quantities L_i^λ, P_i^λ, and S_i^λ will generally not satisfy Eq. (8). Thus, the classification error $E_i^{\dim}$ will be larger and therefore indicate that the point could not clearly be classified as part of a linear, planar or scattered segment of the point cloud.

We will use the classification error (7) in our quantitative experiments in Section 5. Aiming for an as-clear-as-possible classification of points, we pursue as-small-as-possible values of $E^{\dim}$. However, the above discussion depends on the assumptions provided in Proposition 1. In the following we will discuss cases in which these assumptions are not satisfied.

4.2. Degenerate covariance matrix

In practical applications, the assumptions of Proposition 1 are not always satisfied. Note here that the classification error $E_i^{\dim}$ is evaluated on a single point p_i of the point set P. The following reasons can hinder the correct evaluation:

(i) If the point set contains multiple duplicates of a point, more than the sought-for number of neighbors k, all points in the reported neighborhood collapse into a single point equal to the barycenter of the neighborhood. Thus, C_i becomes 0.

(ii) If a point p_i has a flipped normal in comparison to all its neighboring points p_j, the argument x in the weight equation $w_{ij} = \text{sig}_{a,b}^{\cos}(x)$ becomes π and therefore, all weights degenerate to 0 (except when $a = \pi$, which creates uniform weights). This degeneration happens in particular for very small or thin geometries as well as for faulty normal fields.

(iii) Even if the assumptions of Proposition 1 are satisfied, it only states the existence of a suitable parameter $b \in [a, \pi]$. Therefore, choosing parameter b too small can cause all weights in the covariance matrix (4) to degenerate to 0.

In the following evaluation, we prevent case (i) by requiring the point sets to only contain distinct points. Furthermore, we orient the normal field to counteract case (ii). Also, we include the parameter choice $a = b = \pi$ to always have a valid, non-degenerate choice for case (ii). Concerning a too small parameter b, we report a failure in the computation of the error values for the point p_i if $\sum_{\ell=1}^3 \lambda_i^\ell = 0$. By including the choice $a = \pi$ for the parameters, we ensure that each model has at least one correctly evaluated error value $E_i^{\dim}$ at each point $p_i \in P$.

5. Evaluation results

In this section, we present our quantitative evaluation of the weights presented in Eq. (2). For the evaluation, we utilize the classification error $E^{\dim}$ as defined in Eq. (7). Our clean models are taken from a data set described in [27]. The authors provide ten thousand clean and manifold surface meshes, which are obtained by exporting only the boundary of the tetrahedral meshes used in [27]. From these, we randomly select a subset of 1, 000 meshes with uniform probability. Furthermore, we use 100 meshed models each from the real-world object scans provided by [28] and by [29]. Finally, to test the scalability of our approach, we also include the model "Pan et Oursons" from [30].

For all these models, we use the mesh information and its manifold property to obtain oriented face normals. From these, we compute vertex normals and then use these and the vertices as point sets for our experiments. For each such point set P, we consider the parameter sets

$$\mathfrak{A} := \mathfrak{B} := \left\{0, \frac{\pi}{6}, \frac{\pi}{3}, \frac{\pi}{2}, \frac{2\pi}{3}, \frac{5\pi}{6}, \pi\right\}$$

for the choice of a and b, respectively, where we ensure that $a \leq b$. We choose this range as a reasonable trade-off between complexity of the experiments and exploration of the parameter space. Note that in Section 5.5, we consider further parameter values that are rooted in the application scenario considered there. We use the combinatorial neighborhood notion,[2] so that for every pair $(a, b) \in \mathfrak{A} \times \mathfrak{B}$ and every point $p_i \in P$, we calculate its $E_i^{\dim}$ value over the range of k, taken from

$$\mathfrak{K} := \{6, \ldots, 20\}.$$

We assume this range for k, as it reflects typical, heuristic choices for neighborhood sizes in the area of point set processing, see the works discussed in Section 2, in particular [14]. For each point p_i in each point set P, we obtain an optimal parameter triple (a_i^*, b_i^*, k_i^*) as

$$(a_i^*, b_i^*, k_i^*) = \underset{(a,b,k) \in \mathfrak{A} \times \mathfrak{B} \times \mathfrak{K}}{\arg\min} E_i^{\dim}. \tag{9}$$

Following the discussion from Section 4.2, we set $E_i^{\dim} = \infty$ if the covariance matrix C_i for the point $p_i \in P$ degenerates for all parameter choices $(a, b, k) \in \mathfrak{A} \times \mathfrak{B} \times \mathfrak{K}$.

---

[2] For a point $p_i \in P$, we consider the index i as well as the indices of the k nearest neighbors to p_i within P as neighborhood $\mathcal{N}_i$, i.e., $|\mathcal{N}_i| = k + 1$.

(a) Equal Weights: $a = b = \pi$, as in [12].

(b) Cut-Off Weights: $a = b = \arccos(0.9) \approx 0.451$ as in [9], red points show degenerate covariance matrix.

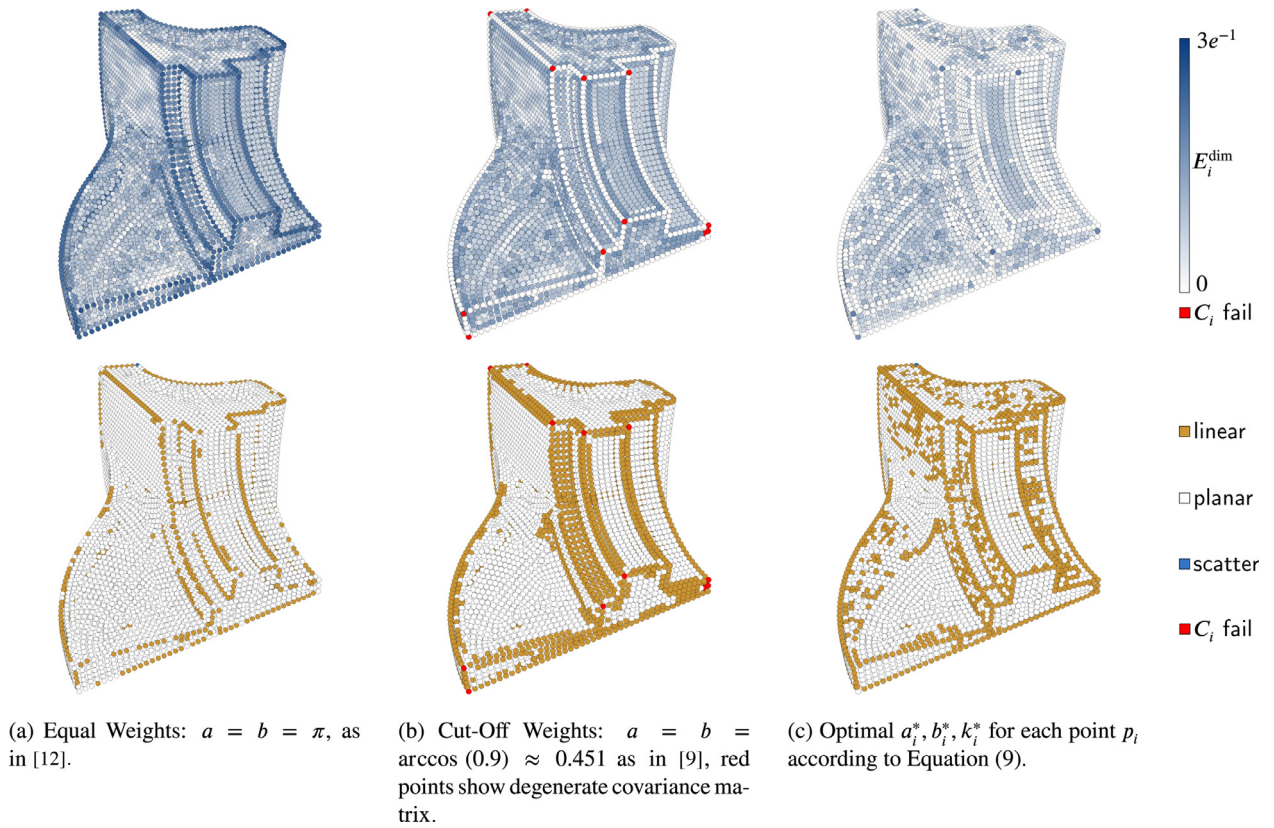(c) Optimal a_i^*, b_i^*, k_i^* for each point p_i according to Equation (9).

Fig. 3. The effect of the different parameters on the fandisk model. The top row shows the classification error $E_i^{\dim}$ from Eq. (7) for each point of the model, from low (white) to high error (dark blue). Note how the optimal weights from Eq. (9) have drastically reduced error in comparison to both equal weights (used by [12]) and sharp cut-off weights (used by [9]). The red points indicate elements of the point set, for which the covariance matrix from Eq. (4) degenerates given the chosen weights. Bottom row shows a feature classification according maximum value out of linearity (orange), planarity (white), and scattering (dark blue) as defined in Eq. (6). Note how equal weights fail to consistently identify edge structures. The cut-off weights manage to identify planar areas well while over-pronouncing edge structures. These are identified well by the weights from Eq. (9). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

See Fig. 3 for an illustration of the classification error $E^{\dim}$ on the fandisk geometry as well as for a comparison of different parameter choices (a, b). The top row shows the classification error $E_i^{\dim}$ from Eq. (7) on each point of the geometry, colored from low error (white) to large error (dark blue). Note that when fixing parameters (a, b), it is possible that the covariance matrix C_i degenerates for every choice $k \in \mathfrak{K}$. This happens for the specific choice $a = b = \arccos(0.9)$ as used in [9]. We have colored the respective points red. The optimal triple from Eq. (9) achieves significantly lower classification error $E_i^{\dim}$ than the equal weights of [12] or the cut-off weights of [9]. The observed fluctuation in planar areas is due to (a) the utilization of combinatorial neighborhoods, which do not always provide symmetrically shaped neighborhoods on a synthetic geometry like the fandisk, as well as to (b) the sensitivity of $E_i^{\dim}$ to slight changes in the covariance matrix.

The bottom row of Fig. 3 shows a feature classification according to the maximum value out of linearity (orange), planarity (white), and scattering (dark blue) as defined in Eq. (6). Note how the equal weights of [12] classify almost all elements as planar and fail to identify edge structures. In contrast, the cut-off weights of [9] identify all edges, but over-pronounce them. The optimal weight choice from (9) takes a middle ground between these two extremes, on the cost of identifying several clearly planar points as linear. Again, this stems to a certain extend from the run a synthetic geometry. Observe that the equal weights and the optimal weights do identify scattered points (one example being the topmost corner of fandisk) while the cut-off weights rather fail to create a covariance matrix at corner points.

The images in Fig. 3 summarize our following experiments. In order to compare with the findings of [12], we compute the classification error $E_i^{\dim}$ for each point of every point set of the three chosen model repositories [27–29] as well as of the single, large model from [30]. In the following we report and interpret our findings.

5.1. Global (a, b, k) analysis

We analyze the total amount of (a, b, k) choices for all model repository selections. Here, we count all points of all point sets with their respective optimal parameter triple (a^*, b^*, k^*). The corresponding four global histograms for the three model repositories and the model from [30] are given in Fig. 4. There, each point of each geometry contributes one unit in the histograms, which report the number of points that choose a parameter combination (a, b), with a on the x-axis and b on the y-axis. Additionally, each such bar is colored according to the chosen optimal neighborhood size k, from the lowest at the bottom to the highest at the top. In summary, the classification error acts similar on all data sets, i.e., in the comparison between clean and real-world models.

On the large scale of $1,000$ point sets with a total of $7,213,429$ points (Fig. 4(a)), we observe, that on average, a small choice for parameter a and a similarly choice for parameter b are preferred. This can be interpreted to say that it is desirable to take only normals into account that exhibit a small deviation. In particular, Fig. 4(a) suggests that the majority of points from the clean models choose a neighborhood without any room for normal
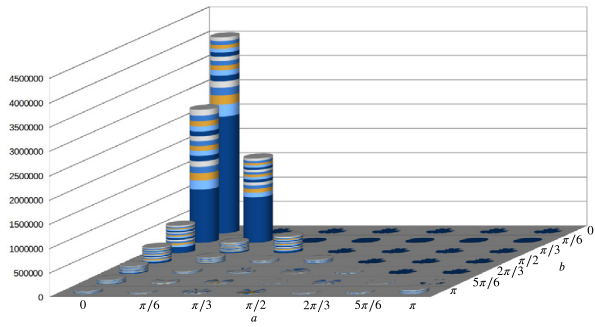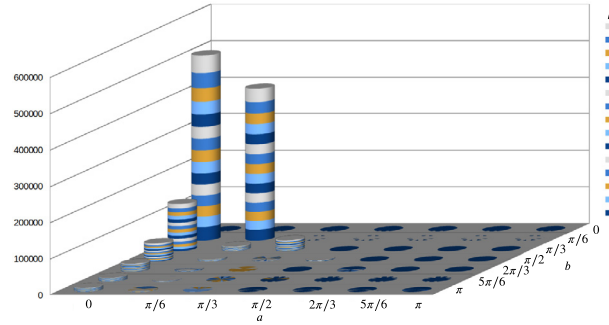
(a) Applied to 1,000 geometries randomly selected from the data set used in [27], with 7,213,429 total points.
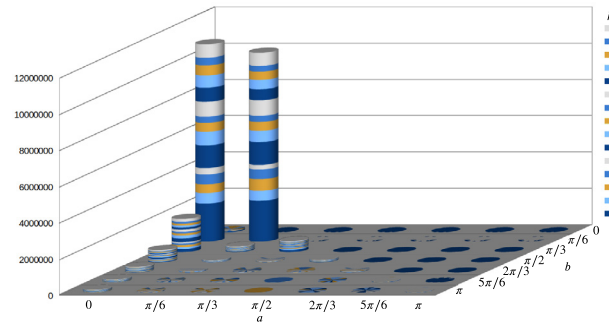
(b) Applied to 100 geometries taken from [28], with 25,929,256 total points.
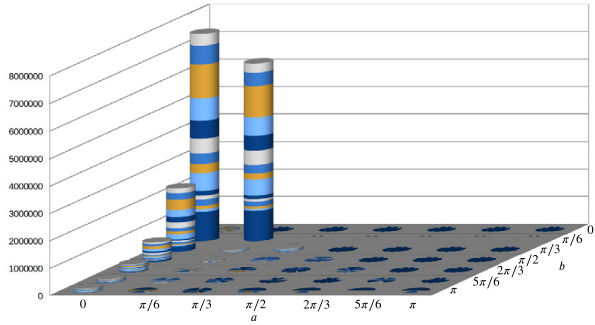
(c) Applied to the 100 geometries from the data set presented in [29], with 17,918,016 total points.

(d) Applied to the "Pan et Oursons" scan from [33], with 1,199,992 total points.

Fig. 4. Histograms of preferred sigmoid parameters (a^*, b^*, k^*) (Eq. (9)) with respect to minimal error values for E^{dim} (Eq. (7)) over the range $\mathfrak{K}$ when applied to several large model repositories. Each point of the respective point set(s) corresponds to one unit in the histogram. Additionally, each such bar is colored according to the chosen optimal neighborhood size k, from the lowest at the bottom to the highest at the top.

deviation ($a = b = 0$). This is one notable difference to the histograms on scanned models, Figs. 4(b) to 4(d), where these drastic weights are almost never chosen.

It is particularly noteworthy that almost no points chose equal weights $a = b = \pi$ which highlights the benefit of our approach over that chosen by [12]. Furthermore, choosing a sharp cut-off along the lines of [9], by $a = b = \pi/6$, occurs for about a quarter of the points. However, about 38% rather go with a softer decrease by choosing $a = 0, b = \pi/6$. A localized, i.e., model-depended, discussion about the possibility to increase a and b for better results is given in the upcoming section.

In terms of scanned real-world models (Figs. 4(b) to 4(d)), we analyzed 100 point sets from each [28,29] and one large model from [30]. In comparison to the clean models, we do observe a different behavior. Namely, while small values for a and b are still favored, the choice of $a = 0$, which was most prominent on clean models, is almost never made for scanned models. The chosen weights indicate that mostly neighborhoods with a normal deviation of up to $\pi/6$ are taken into account. These are either all weighted uniformly ($a = \pi/6$) or with gradually deteriorating weights ($a = 0$). We interpret the parameter b to reflect the noise components caused by the acquisition process. Therefore, choosing the lowest possible choice of b causes several points $p_i \in P$ to have degenerate covariance matrices C_i, independent of the chosen neighborhood size k. See also the following section for a more detailed discussion of this.

In conclusion, we see that weight-determination generally favors a narrow window between parameters a and b. This corresponds to using a neighborhood with an overall small normal deviation. The value b however depends on the geometry. Clean models mostly attain smaller error values for very small values of b, whereas real-world models require slightly larger of b to obtain non-degenerate covariance matrices. All models from all

repositories have in common that they almost never report equal weights as preferred weight assignment. Hence, when regarding the classification error E_i^{dim}, the equal weighting scheme of [12] is inferior to the family of weights presented here. This becomes obvious when comparing the values obtained from our experiments, see Table 1. The classification error computed with our weights (9) has lower minimum, average, maximum, and standard deviation than the error computed with the equal weights of [12].

Sharp cut-off weights are only chosen as optimal weighting by a subset of the real-world scans. As [9] used sharp cut-off weights in the context of denoising, our results hint that this weight set might be beneficial in the presence of noise. However, for about 75% of the scanned models, when considering the classification error E_i^{dim} our weighting family still chooses weights superior to the cut-off weights used by [9].

5.2. Local (a,b) analysis

In this section, we will discuss the (a^*, b^*) choices presented in the previous section from a local, i.e., point-set-dependent, perspective. The respective results are presented in Table 2. There, the first row corresponds to the clean models from [27] while the other three rows correspond to the scanned real-world models from [28–30]. The columns present information about the amount of points accepting minimal value $a = 0$, allowing ($a-$) or forbidding ($\neg a-$) a decrease of a, accepting minimal value $b = a$, and allowing ($b-$) or forbidding ($\neg b-$) a decrease of b. In this scheme, the columns $\neg a-$ and $\neg b-$ denote the percentage of those points for which a decrease of the respective parameter results in a degenerate covariance matrix C_i, see Section 4.2. Observe that we cover all possible cases. For easy comparability, we provide the respective case numbers in percent, with the total

Table 1

Quantitative comparison of the classification error E_i^{dim} computed over different model repositories with weights by [12] and our weights (9). Note that our weighting scheme always obtains lower minimal, average, and maximum error as well as a lower standard deviation.

		$\min_i(E_i^{\text{dim}})$	$\frac{1}{N}\sum_{i=1}^{N} E_i^{\text{dim}}$	$\max_i(E_i^{\text{dim}})$	$\text{sd}_i(E_i^{\text{dim}})$
Clean [27]	Weights [12]	0	0.6309891	1.071584	0.1896867
72,13,429 points	Our (9)	0	0.2477968	1.012825	0.1882686
Scanned [28]	Weights [12]	$2.946194 \cdot 10^{-8}$	0.3348262	1.071796	0.1967012
2,59,29,256 points	Our (9)	0	0.2526083	0.9649821	0.1569379
Scanned [29]	Weights [12]	0.001344446	0.2497551	1.048021	0.1144664
1,79,18,016 points	Our (9)	0	0.2109502	0.9379431	0.09631257
Scanned [30]	Weights [12]	0.003978099	0.3827366	0.9805029	0.1456077
11,99,992 points	Our (9)	0	0.3200202	0.7774453	0.1337365

Table 2

Distribution of (a^*, b^*) choices into the three cases of (a) an attained minimum ($a = 0$, $b = a$), (b) a possible decrease of the parameter without failure ($a-$, $b-$), and (c) impossibility of decreasing the parameter because it would cause a degenerate covariance matrix ($\neg a-$, $\neg b-$).

	$a = 0$	$a-$	$\neg a-$	$b = a$	$b-$	$\neg b-$	#points
Clean [27]	60.19%	39.81%	0%	36.80%	25.49%	37.72%	7,213,429
Scanned [28]	54.30%	45.70%	0%	43.50%	14.21%	42.29%	25,929,256
Scanned [29]	62.44%	37.56%	0%	36.86%	21.03%	42.11%	17,918,016
Scanned [30]	60.70%	39.30%	0%	37.33%	20.46%	42.21%	1,199,992

number of points for the respective repository given in the last column.

Having all values in one chart, we directly observe the behavior assessed for parameter a in the previous section. There, we stated that especially in the case of clean models, an as-small-as-possible value for a is favorable over larger values for a. Indeed, Table 2 confirms this statement, as almost[3] none of the points forbid a decrease of parameter a (cf. column $\neg a-$). This justifies the small values for a attained in the real-world scenarios presented in Figs. 4(b) to 4(d) when compared to the values of a attained in the clean scenarios in Fig. 4(a). Semantically, this opts for including just enough neighbors in the computation to make it feasible, i.e., to prevent a degenerate covariance matrix, but focus on those that are as similar as possible with regard to the normal field.

The reported numbers on the parameter b also support the observation drawn before. It is chosen to be as small as possible, i.e., as close to the chosen a without creating a degenerate covariance matrix. Over all repositories, b is chosen to create a sharp cut-off ($b = a$) in about 36% of the considered points. A notable exception is the scanned data set [28], which allows for 43.5% of the points to choose a sharp cut-off. This is possibly due to lower noise levels and different geometry types in this data set when compared to the other scanned data sets [29,30]. Furthermore, in about 42% of the scanned points and 37% of the clean points, b is at least chosen to be as close as possible to a, i.e., the weighting scheme is chosen to be as close as possible to a sharp cut-off, which cannot be realized because a further decrease of b would cause a degenerate covariance matrix. These observations justify the general weighting choice of sharp cut-off, as chosen by [9], although the particular chosen values only prove to be most effective in about one fourth of all models from the data set used here.

Summarizing the global and local analysis of the parameter choices (a^*, b^*), we draw the following conclusions:

- The utilized classification error favors weight determination with as-small-as-possible values for both parameters a and b. That is, only points with as-similar-as-possible normals are considered, but out of these, all are allowed to influence the computation as evenly as possible.

- Equal weights ($a = b = \pi$), as used by [12], are never chosen as optimal parameters to obtain a minimal classification error E_i^{dim}.
- Sharp cut-off weights as widely used in the literature, e.g., in [9], attain minimal classification error for 36.8% of the clean points and for up to 43.5% of the scanned points. This proves their relevance in particular for real-world scenarios.

5.3. Global k analysis

As stated in the beginning of Section 5, for each point in the utilized point sets, we also obtain a preferred neighborhood size $k^* \in \mathfrak{K}$ yielding smallest classification error E_i^{dim} among all choices (Eq. (9)). In Fig. 5, we present a histogram plotting this data, i.e., for each neighborhood size $k \in \mathfrak{K}$, we show what percentage of points from the respective model repository use this k.

Note that the plot for the clean models shows a favor for an as-small-as-possible neighborhood size k over larger neighborhoods. In contrast, the scanned models show a different behavior. Whereas the repository [28] also attains its peak at $k = 6$, it is more equally distributed among the whole range, with a notable second peak at $k = 11$. The models in [29], however, exhibit an almost Gaussian bump around their maximal value $k = 18$. Finally, the results for the model chosen from [30] are almost uniform over the entire range, with a slight increase for growing values of k.

In order to investigate scalability effects, we have included one significantly larger model from [30] in our analyses. Note that the general observations as made in Fig. 4, Table 1, Table 2, and Fig. 5 particularly hold for this model. Furthermore, despite the fact that [12] focuses on these large-scale models, our weighting approach (9) still provides smaller classification errors as indicated in Table 1. Thus, we cannot report any scaling issues.

For the clean models, we obtain an average neighborhood size of $\bar{k} = 10.55$ with a standard deviation of $\sigma = 4.77$. For the scanned models, those quantities are:

- $\bar{k} = 12.09489$, $\sigma = 4.618431$ [28],
- $\bar{k} = 14.22883$, $\sigma = 4.487704$ [29],
- and $\bar{k} = 13.54247$, $\sigma = 4.398983$ [30].

These findings suggest that variable neighborhood sizes yield smaller error values with regard to the classification error E_i^{dim}.

---

[3] There are $< 0.01\%$ for each scanned repository that forbid a decrease, which is not shown here due to rounding. In contrast to Proposition 1 (stating independence of the choice of a, rounding issues within the application prevented a decrease of a.
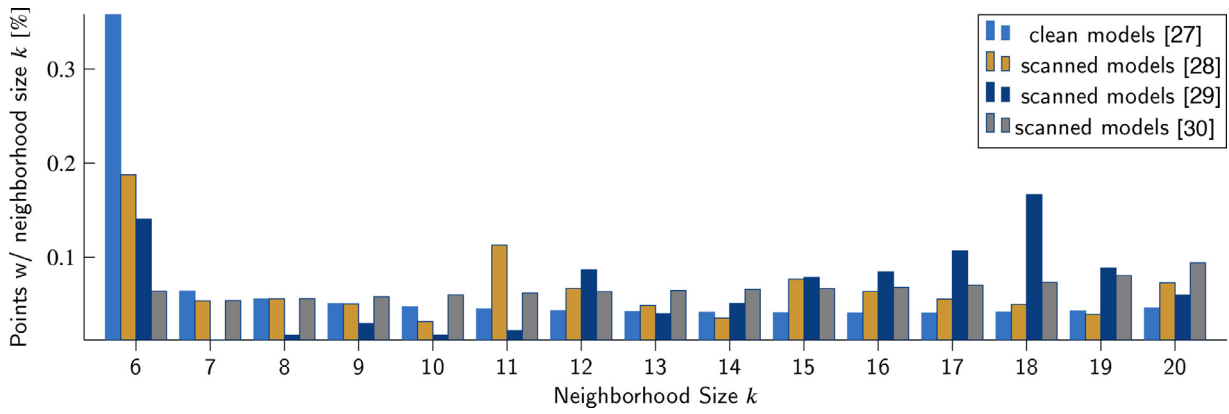
Fig. 5. Histogram of preferred neighborhood sizes k^* with respect to minimal error value $E_i^{\dim}$. To ensure compatibility over the different data repositories, we normalize by the total number of points and report the percentage of points choosing the respective neighborhood size.

If a global neighborhood size has to be chosen, then the average values provided by this analysis serve as reasonable choices, as they provide a good trade-off between a fixed neighborhood size and a low classification error. In order to further investigate the benefit of varying neighborhood sizes, in the following section, we turn to a point-set-dependent perspective.

5.4. Local k analysis

We will now consider the standard variation of the neighborhood sizes taken over a single model for $E^{\dim}$. We aim to better understand and investigate the hypothesis formulated above, i.e., the statement that a variable neighborhood size contributes to lower classification error.

In order to interpret the neighborhood sizes, we consider a box–whisker plot over all standard deviations within the respective models in Fig. 6. That is to say, the boxes indicate the first, second (median), and third quartile of the standard deviations of neighborhood sizes for the indicated model repository. In particular, most approaches in the literature use—and are evaluated on—a setting with a fixed neighborhood size k. In our analysis, this would correspond to a standard deviation around 0, indicating no or small changes to the neighborhood size within a geometry. However, it is obvious from Fig. 6 that all standard deviations are located well away from 0. Even considering the minima, i.e., the lower whiskers of the boxes, they reside at 2.7, 4.1, and 4.4, respectively, indicating that at least these small variations in neighborhood size are necessary to minimize the classification error. Note that the variation of neighborhood size is most notable for the clean model repository, where the standard deviation of chosen neighborhood sizes goes up to 5.46. This is in contrast to the scanned data from [29], where the models are not very diverse, which is reflected in the almost uniform standard deviation of the chosen neighborhood sizes.

In summary, from the global and local analysis of the obtained neighborhood sizes k, we draw the following conclusions:

- All standard deviations lie well above 0, i.e., the considered classification error favors variable neighborhood sizes over constant-size neighborhoods.
- This behavior is more pronounced for scanned models [28] and [29] than for clean models [27].
- The classification error favors smaller neighborhood sizes for clean models, however for scanned models this behavior is not preserved.

5.5. Application scenario

Building on the observation from the previous section that varying neighborhood sizes can contribute to better performance and in order to evaluate our proposed methodology in an application scenario, we turn to the normal filtering stage of the point set denoising algorithm proposed in [9]. This first stage is part of a larger, iterative process of three stages that removes noise from an input geometry. We focus on the first stage to not have the effect of our weighting scheme be confounded by procedures within the more complex pipeline (we provide the names of the parameters of the algorithms in brackets in the following). In each iteration (parameter p), a weighted covariance matrix is built. The algorithm is using a sharp cut-off weight function (parameter ρ), optimizes the eigenvalues of the covariance matrix (parameter τ), and uses those to update the respective point normals afterwards.

For the experiment, we followed the experimental pipeline of the original article [9]. Namely, we took the four geometries (Cube, Fandisk, Octahedron, Rockerarm) as discussed in the original publication [9], together with three more geometries (Bearing, Sharp Sphere, Fertility), see Fig. 7. The models were given as meshes and provided the oriented point normals, as they were obtained from the mesh information as weighted vertex normals that served as ground truth normals in the experiment. Afterwards, we applied Gaussian noise in normal direction with amplitudes 0.3ℓ and 0.6ℓ, where ℓ denotes the mean of all distances from points to their respective six nearest neighbors. To measure the deviation from the ground truth normals, we make use of the mean squared error (MSE) given as

$$\text{MSE}(N, \tilde{N}) = \frac{1}{n} \sum_{i=1}^{n} \|n_i - \tilde{n}_i\|_2^2, \tag{10}$$

where $N = \{n_i \mid p_i \in P\}$ are the ground truth and $\tilde{N} = \{\tilde{n}_i \mid p_i \in P\}$ are the changed normals of the geometry P.

The default values for the normal filtering were taken from [9] and the parameters (p, ρ, τ) are set as follows:

- Cube with $(150, 0.95, 0.3)$,
- Fandisk with $(150, 0.9, 0.3)$,
- Octahedron and Rockerarm with $(80, 0.9, 0.25)$,
- and Bearing, Sharp Sphere, and Fertility with $(100, 0.9, 0.25)$.

For the allowed neighborhood range, we first let the default normal filtering of [9] identify the neighborhoods. Then, for each point p_i, we took the number of neighbors k_i identified and let $\mathfrak{K}_i := \{k_i - 10, \ldots, k_i + 10\}$. If this caused negative values, we omitted them. Afterwards, each point chose a neighborhood according to the least error (Eq. (7)) before and in each iteration.
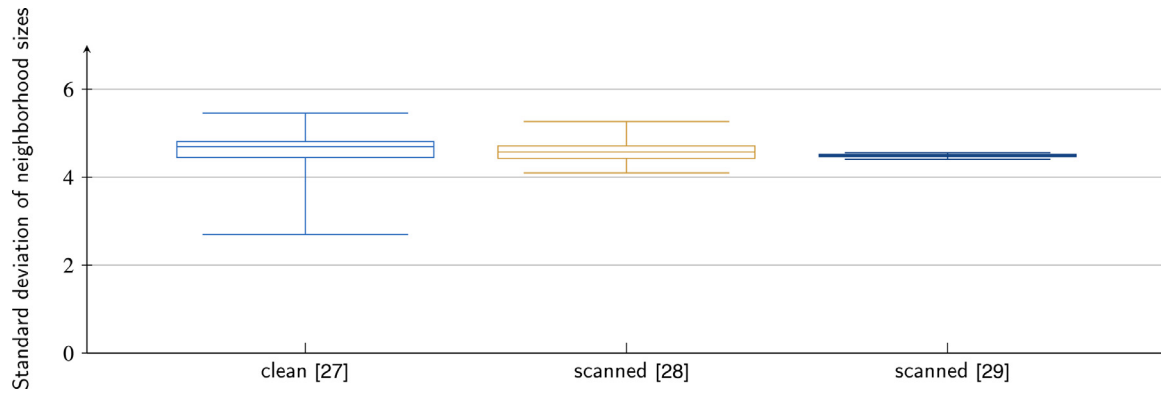
Fig. 6. Box–whisker plot for the standard deviations obtained by the different models. Each model contributes its own standard deviation as a data point for the diagram. Therefore, the leftmost column represents 1,000 data points (from [27]), the center column represents 100 data points (from [28]), and the rightmost column represents 100 data points (from [29]).
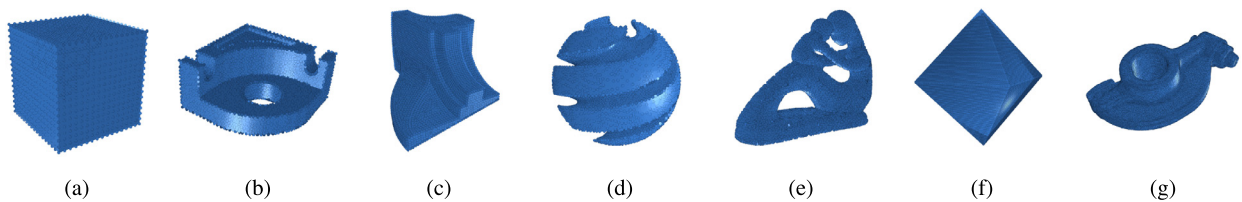

| (a) | (b) | (c) | (d) | (e) | (f) | (g) |

Fig. 7. Geometries from left to right with number of vertices in brackets: Cube (1,906), Bearing (3,475), Fandisk (6,475), Sharp Sphere (8,354), Fertility (9,239), Octahedron (16,395), and Rockerarm (24,106).

We set parameters $a = b = \arccos(\rho)$ w.r.t. the processed geometry, as this algorithm is tailored to benefit from a sharp distinction. In the cases where the covariance matrix degenerates as all weights become 0, we use the default neighborhood size and equal weights for such points.

This allows us to study the effect of including our weighting scheme into a larger application by comparing to the results obtained by [9]. Furthermore, we can compare both the original results of [9] and the results of our enhanced weighting pipeline to the ground truth normals provided by the noiseless models. In Table 3, the MSE results are shown, where MSE(Noise) compares ground truth and noisy input, MSE [9] compares ground truth and the result of [9] with its default parameters, and MSE(ours) compares ground truth and the result of the enhanced normal filtering pipeline.

From Table 3, an immediate observation is that allowing individual neighborhood ranges reduces the overall error compared to the default values used in the normal filtering. This becomes obvious as all values in the column MSE(ours) are smaller than those in column MSE [9]. Thus, the algorithm benefits from individual neighborhood ranges as discussed in Section 5.4, instead of setting one global neighborhood size parameter.

One interesting observation can be made by incorporating the MSE(Noise) values into the comparison. For most of the models considered, these values outperform the normal filtering results. To explain this, consider that the point normals are influenced by the mesh properties, i.e., the faces and their areas affect the normals provided as ground truth. The filtering procedures thus sometimes worsen the results as it tries to clearly arrange normals corresponding to points which represent a planar area, whereas points lying on (sharp) edges in a mesh receive normals neither belonging to the areas meeting at that edge. This points towards future work and improvements of [9], but is irrelevant in the context of this paper as the main message to be taken from these experiments is that utilizing optimized neighborhood selection over a wider range of k does positively influence the performance of the algorithm.

6. Conclusion

In this article, we investigated a family of weights (Eq. (2)) for point set processing. These weights are based on the normal similarity. The family includes common choices such as equal weights or sharp cut-off weights at a given threshold. Furthermore, we presented an evaluation model for neighborhood weights based on a Shannon entropy classification error (Eq. (7)). We have performed a large-scale evaluation of our weight family on four data sets. The first set consisted of 1,000 clean surface meshes from the work of [27]. The second and third set consisted of 100 real-world scans taken from each [28] and [29]. Additionally, we included a scanned model from [30] with over one million points.

A statistical analysis revealed that the optimal weight parameters should lead to a neglect of non-similar normals, yet include mid-range normal points with a low weight. Specifically, equal weights, as used in the literature discussed in Section 2 and in particular in [12] do not obtain minimal error values. Furthermore, sharp cut-off weights as used, e.g., by [9] do perform well on certain scanned models, but are also generally inferior to more flexible weighting terms. Finally, it became obvious in the evaluation that neighborhood sizes have to be variable over a point set as only these variable sizes attain minimal error values. The potential of this variability of neighborhood sizes was shown by incorporating point-wise neighborhood size ranges within a normal filtering stage in a point denoising algorithm [9], yielding smaller mean squared errors compared to the original pipeline.

While this article addresses a variety of possible weighting choices and neighborhood sizes, to cover the most widely used versions from the literature, several aspects are left as future work. Further research consists of running the large-scale analysis on a broader range of neighborhood sizes, comparable to [12].

Not only with the publication of the versatile *PointNet* architecture, machine- and deep learning techniques gradually conquered the realm of point set processing [31]. Subsequently, a wide variety of publications arose that tackle several problems

Table 3

Mean squared errors (MSE) for several models corrupted by noise 0.3ℓ and 0.6ℓ with ℓ being the mean distance of all distances among six nearest neighbors. Our approach outperforms that of [9] in all examples considered.

	0.3ℓ			0.6ℓ		
	MSE(Noise)	MSE [9]	MSE(ours)	MSE(Noise)	MSE [9]	MSE(ours)
Cube	0.018171157	0.004720763	**0.003817538**	0.067785183	0.041014649	**0.017185731**
Bearing	**0.019149792**	0.092202990	0.06311642	**0.087963467**	0.105885719	0.099716185
Fandisk	**0.015985698**	0.047105692	0.033043562	0.059076477	0.067027229	**0.052858261**
Sharp sphere	**0.033954089**	0.152198571	0.124993406	**0.091947551**	0.162334741	0.143099697
Fertility	**0.134043694**	0.220368116	0.197418828	**0.232151395**	0.25364564	0.25100736
Octahedron	0.287851891	0.282346769	**0.275334065**	0.304714745	0.277547626	**0.271626537**
Rockerarm	**0.017150895**	0.081370198	0.067150116	**0.063687905**	0.084070141	0.078171864

related to point set processing. These touch on multiple areas discussed in this paper. For instance, the *PointCleanNet* architecture was designed for denoising and outlier removal of point sets [32]. Another representative of machine learning technology for point sets is the *NormNet* architecture, which derives point-wise normal estimations for three-dimensional point sets [33]. A problem naturally approached using normal information on a point cloud is (semantic) segmentation. In a sense, our optimized neighborhood weights do segment the point set into several, small parts with consistent normal information. In contrast, large-scale segmentation approaches via machine learning are available, e.g., utilizing edge-convolution networks [34]. All these machine learning approaches have in common that they depend on large sets of well labeled training data. In its generality, our approach can add to these developments in many ways. Aside from improving the (semi-)automated generation of training data, it can support the selection of neighborhood weights and sizes that serve as architecture input. Additionally, the direct incorporation of normal information on the neighborhood-level supports topological consistence, outlier resistance, and coherence of both the considered point set as well as the algorithmic procedure applied to it. Investigating these potentials of our methodology in the machine learning context is, however, left for future work.

As a closing remark, we would like to point out that several algorithms developed within the geometry processing community are heavily dependent on one or more parameters. These parametric values are often fine-tuned during an experimentation phase that is run on a small and limited data set. Our large-scale analysis in this paper highlights the importance of a systematic setup for parameter evaluation. In particular, we were able to show that the usually globally chosen neighborhood size parameter yields better results when used within a bilateral weighting scheme that incorporates normal information and varying neighborhood sizes. We hope that this work encourages further research on the applicability of algorithms and their parameters "in the wild" to ensure applicability and robustness of the developed methods.

CRediT authorship contribution statement

Martin Skrodzki: Conceptualization, Methodology, Software, Evaluation. **Eric Zimmermann:** Conceptualization, Methodology, Software, Evaluation.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] Levoy M, Whitted T. The use of points as a display primitive. Technical report, University of North Carolina; 1985.

[2] Levoy M, Pulli K, Curless B, Rusinkiewicz S, Koller D, Pereira L, et al. The digital Michelangelo project: 3D scanning of large statues. In: Proceedings of the 27th annual conference on computer graphics and interactive techniques; 2000. p. 131–44.

[3] Boehnen C, Flynn P. Accuracy of 3D scanning technologies in a face scanning scenario. In: IEEE fifth international conference on 3D digital imaging and modeling. 2005, p. 310–7.

[4] Buck U, Naether S, Braun M, Bolliger S, Friederich H, Jackowski C, et al. Application of 3D documentation and geometric reconstruction methods in traffic accident analysis with high resolution surface scanning, radiological MSCT/MRI scanning and real data based animation. Forensic Sci Int 2007;170(1):20–8.

[5] Bellekens B, Spruyt V, Berkvens R, Weyn M. A survey of rigid 3D point-cloud registration algorithms. In: AMBIENT 2014: The fourth international conference on ambient computing, applications, services and technologies, August 24-28, 2014, Rome, Italy. 2014, p. 8–13.

[6] Alexa M, Behr J, Cohen-Or D, Fleishman S, Levin D, Silva CT. Point set surfaces. In: VIS'01: Proceedings of the conference on visualization '01. IEEE Computer Society; 2001, p. 21–8.

[7] Park MK, Lee SJ, Lee KH. Multi-scale tensor voting for feature extraction from unstructured point clouds. Graph Models 2012;74(4):197–208.

[8] Skrodzki M, Jansen J, Polthier K. Directional density measure to intrinsically estimate and counteract non-uniformity in point clouds. Comput Aided Geom Design 2018;64:73–89. http://dx.doi.org/10.1016/j.cagd.2018.03.011, URL: http://www.sciencedirect.com/science/article/pii/S0167839618300256.

[9] Yadav SK, Reitebuch U, Skrodzki M, Zimmermann E, Polthier K. Constraint-based point set denoising using normal voting tensor and restricted quadratic error metrics. Comput Graph 2018;74:234–43. http://dx.doi.org/10.1016/j.cag.2018.05.014, URL: http://www.sciencedirect.com/science/article/pii/S0097849318300797.

[10] Skrodzki M. Neighborhood data structures, manifold properties, and processing of point set surfaces. (Ph.D. thesis), Berlin, Germany: Freie Universität Berlin; 2019.

[11] Demantké J, Mallet C, David N, Vallet B. Dimensionality based scale selection in 3D lidar point clouds. In: The international archives of the photogrammetry, remote sensing and spatial information sciences, vol. XXXVIII-5/W12; 2011. p. 97–102.

[12] Weinmann M, Jutzi B, Mallet C. Semantic 3D scene interpretation: a framework combining optimal neighborhood size selection with relevant features. ISPRS Ann Photogramm Remote Sens Spat Inf Sci 2014;II-3:181–8.

[13] Pauly M, Gross M, Kobbelt L. Efficient simplification of point-sampled surfaces. In: Proceedings of the conference on visualization'02. IEEE Computer Society; 2002, p. 163–70.

[14] Pauly M, Keiser R, Kobbelt L, Gross M. Shape modeling with point-sampled geometry. ACM Trans Graph 2003;22, 3:641–50.

[15] Floater MS, Reimers M. Meshless parametrization and surface reconstruction. Comput Aided Geom Design 2001;18(2):77–92.

[16] Brodu N, Lague D. 3D terrestrial lidar data classification of complex natural scenes using a multi-scale dimensionality criterion: Applications in geomorphology. ISPRS J Photogramm Remote Sens 2012;68:121–34.

[17] Lipman Y, Cohen-Or D, Levin D. Error bounds and optimal neighborhoods for MLS approximation. In: Proceedings of the fourth eurographics symposium on geometry processing. Eurographics Association; 2006, p. 71–80.

[18] Levin D. The approximation power of moving least-squares. Math Comp 1998;67(224):1517–31.

[19] Levin D. Mesh-independent surface interpolation. In: Brunnett G, Hamann B, Müller H, Linsen L, editors. Geometric modeling for scientific visualization. Springer; 2004, p. 37–49.

[20] Sober B, Levin D. Manifolds' projective approximation using the moving least-squares (MMLS). 2016, CoRR arXiv:1606.07104.

[21] Mitra NJ, Nguyen A, Guibas L. Estimating surface normals in noisy point cloud data. Int J Comput Geom Appl 2004;14(04n05):261–76.

[22] Linsen L, Prautzsch H. Local versus global triangulations. In: Proceedings of EUROGRAPHICS, vol. 1; 2001. p. 257–63.

[23] Hoppe H, DeRose T, Duchamp T, McDonald J, Stuetzle W. Surface reconstruction from unorganized points. In: Proceedings of the 19th annual conference on computer graphics and interactive techniques. ACM; 1992, p. 71–8.

[24] Belton D, Lichti DD. Classification and segmentation of terrestrial laser scanner point clouds using local variance information. Int Arch Photogramm Remote Sens Spat Inf Sci 2006;36(5):44–9.

[25] Shannon C. A mathematical theory of communication. Bell Syst Tech J 1948;27:379–423.

[26] Marler MR, Gehrman P, Martin JL, Ancoli-Israel S. The sigmoidally transformed cosine curve: a mathematical model for circadian rhythms with symmetric non-sinusoidal shapes. Stat Med 2006;25(22):3893–904.

[27] Hu Y, Zhou Q, Gao X, Jacobson A, Zorin D, Panozzo D. Tetrahedral meshing in the wild. ACM Trans Graph 2018;37(4):60–1.

[28] Choi S, Zhou Q-Y, Miller S, Koltun V. A large dataset of object scans. 2016, ArXiv:1602.02481.

[29] Bogo F, Romero J, Loper M, Black MJ. FAUST: Dataset and evaluation for 3D mesh registration. In: Proceedings IEEE conf. on computer vision and pattern recognition (CVPR). Piscataway, NJ, USA: IEEE; 2014, p. 3794–801.

[30] Laric O. 2012, URL: http://threedscans.com/.

[31] Qi CR, Su H, Mo K, Guibas LJ. Pointnet: Deep learning on point sets for 3d classification and segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition; 2017. p. 652–60.

[32] Rakotosaona M-J, La Barbera V, Guerrero P, Mitra NJ, Ovsjanikov M. Pointcleannet: Learning to denoise and remove outliers from dense point clouds. Comput Graph Forum 2020;39(1):185–203.

[33] Hyeon J, Lee W, Kim JH, Doh N. NormNet: Point-wise normal estimation network for three-dimensional point cloud data. Int J Adv Robot Syst 2019;16(4):1729881419857532.

[34] Contreras J, Denzler J. Edge-convolution point net for semantic segmentation of large-scale point clouds. In: IGARSS 2019 - 2019 IEEE international geoscience and remote sensing symposium. 2019, p. 5236–9. http://dx.doi.org/10.1109/IGARSS.2019.8899303.

Martin Skrodzki has studied Mathematics and Computer Science at TU Dortmund University (Germany), Texas A&M International University (USA), and Freie Universität Berlin (Germany). He obtained his Dr. rer. nat. in 2019 at Freie Universität Berlin under the supervision of Prof. Polthier (Freie Universität Berlin) and Prof. Levin (Tel Aviv University, Israel). He has been a postdoctoral scholar at the Institute for Computational and Experimental Research in Mathematics (ICERM, Brown University, USA) and in the RIKEN Interdisciplinary Theoretical and Mathematical Sciences Program (RIKEN, Japan). He is currently a DFG-Walter-Benjamin postdoctoral fellow at the Computer Graphics and Visualization (CGV, TU Delft, the Netherlands).

Eric Zimmermann has studied Mathematics at Freie Universität Berlin (Berlin, Germany). He currently is a doctoral candidate in the group "Mathematical Geometry Processing" under the guidance of Prof. Polthier (Freie Universität Berlin, Germany), and he is part of the project C05 of the collaborative research cluster SFB Transregio 109 called "Discretization in Geometry and Dynamics" dealing with computational and structural aspects of point sets.