Response to Reviewer 1 Comments

Major comments.

Point 1: Please give a formal definition of what is the correct normal of a feature point. As you are aware there exist a lot of definitions of what a proper normal for a feature point is.

Response 1: On this issue, I would like to explain in three aspects:
1) In three dimensions, a surface normal, or simply normal, to a surface at point p is a vector perpendicular to the tangent plane of the surface at P. The default definition of a correct normal in our article, as in much of the literature ([Zhou et al.][Guerrero et al.],[Zhang et al.],[Li et al.]…), is the definition of a normal in the general geometric sense. Therefore our original manuscript, like most of the rest of the literature, does not purposely explain it.

Zhou, J., Jin, W., Wang, M., Liu, X., Li, Z., & Liu, Z. (2022). Fast and accurate normal estimation for point clouds via patch stitching. Computer-Aided Design, 142, 103121.
Guerrero, P., Kleiman, Y., Ovsjanikov, M., & Mitra, N. J. (2018, May). Pcpnet learning local shape properties from raw point clouds. In Computer graphics forum (Vol. 37, No. 2, pp. 75-85).
Zhang, J., Cao, J., Liu, X., Chen, H., Li, B., & Liu, L. (2018). Multi-normal estimation via pair consistency voting. IEEE transactions on visualization and computer graphics, 25(4), 1693-1706.
Li, B., Schnabel, R., Klein, R., Cheng, Z., Dang, G., & Jin, S. (2010). Robust normal estimation for point clouds with sharp features. Computers & Graphics, 34(2), 94-106.

2) In order to avoid unnecessary ambiguity, we have cited the literature of Zhang et al (i.e., references in the manuscript [25]) as requested by the reviewer, and the definition of normal is given in this article. Both single normal and multi-normal are included, and as far as I know this is the definition of normal currently accepted by most researchers.

3) The test dataset in our experiments comes with a real normal(i.e., Ground Truth) from PCPNet and PCV.
Guerrero, P., Kleiman, Y., Ovsjanikov, M., & Mitra, N. J. (2018, May). PCPNet learning local shape properties from raw point clouds. In Computer graphics forum (Vol. 37, No. 2, pp. 75-85).
Zhang, J., Cao, J., Liu, X., Chen, H., Li, B., & Liu, L. (2018). Multi-normal estimation via pair consistency voting. IEEE transactions on visualization and computer graphics, 25(4), 1693-1706.

Point 2: Several papers use also PCA on the normal information as in equation (3).

Response 2: Thank you for the comments on the paper. Here we would like to explain the meaning of Eq. 3 in this paper from two aspects.
1) In order to make the initial normals of smooth areas have a certain accuracy, as well as preparation for the subsequent processing, based on the weighted principal component analysis (WPCA) [13],

our NERL take advantage of M-estimates of location [41] instead of the mean to estimate the normals. Fig.2 shows the effect of error comparison between PCA, WPCA and our NERL method for normal estimation. However, we find that although the normals of the points in the smoothed region can be reliably computed by NERL, the normals of the edge points still fail to maintain sharp features. In order to make feature detection more applicable, we divided it into two cases according to the noise scale to deal with separately. Here, for the large-scale noise case, we propose a robust method based on the normal difference. For a given point with a neighborhood, the covariance matrix consists of the normal differences of the pairs of points in the neighborhood, expressed as Eq. 3. We compared the extraction effect between $U = \frac{1}{S}\sum_{p_i \in N_{p_i}} (p_i - \overline{p}_i)(p_i - \overline{p}_i)^T$ (i.e.,PCA) and $U = \frac{1}{S}\sum_{n_j, n_k \in N_{p_i}} (n_j - n_k)(n_j - n_k)^T$, (i.e.,Eq.3). Fig. 3 indicates that our method can still perform accurate feature detection even if under large-scale noise.

2) The size of the three eigenvalues of the covariance matrix U reflects the dispersion degree of the normals in the neighborhood. For smooth regions, the three eigenvalues tend to be zero, and as the enlargement of dispersion degree, the size of the eigenvalues increases accordingly. If we concentrate the beginning of the normals of the neighborhood points to the origin, and the basic rationale of the formula is to treat the end of the normals as points, and then calculate the shape variation of these points, If on a plane, these points will all be in the same position, then the sum of the three eigenvalues will be zero, and on sharper surfaces the sum will be greater.

Point 3: Note that equations (6, 8) are similar in nature to equation (2) of ref.[5]. So mention this and explain the differences of your approach with their approach. In [5] they use this equation to separate feature from non-feature points. Explain why your approach is better than their approach.

Response 3: Thank you for reminding us the improper description on the study. We have revised the description of the manuscript to meet your requirements. Here are some explanations for your other questions.

If we just look at the form of single equations, our Equations 6 and 8 may be similar to Equation 2 in the literature[5]. However, our purpose and process are not exactly the same as [5].

In section 5, we explain how to reliably estimate the normals of the distinguished feature points via the normal mollification methodology(i.e.,Eqs. 6 and 8), which divides into two successive steps: a preliminary normal estimation based on neighborhood segmentation (Eq. 6) and a further normal estimation based on residual under various difficult scenarios (Eq. 8).

More important, after abnormal points' mollification neighborhood is sought, the second-stage normal mollification is designed to estimate normals of these points more finely. Due to the existence of noise, non-uniform sampling, or various complications, some non-feature points are over-identified as feature points such as narrow-band and low sampling density regions, so the normals in some points may be incorrectly estimated. To cope with this kind of issues as well as compensate

for the impact of excluded feature points in the first stage, the second-stage normal mollification is devised.

Furthermore, the above process differs from the idea of removing the boundary points first and then resampling them in literature [5]. It is also important to note that the process in literature [5] changes the position of the points(similar to the point update method) in the actual point cloud model, whereas our method only corrects the normal direction and does not change the spatial position of the original points.

Point 4: In the experiments please use all the methods in the figures not some on one experiment and some in other. All the approaches need to be present.

Response 4: Thank the reviewer for the comments. We have added all methods to the experimental setup based on your suggestions. Also revised Fig. 7 to Fig. 9.

The original manuscript was set up in this way because it took into account the research focus of the different approaches and made trade-offs for different types of experiments, similar to the experimental setup in papers [Zhang et al.] and [Zhou et al.].

Zhang, J., Cao, J., Liu, X., Chen, H., Li, B., & Liu, L. (2018). Multi-normal estimation via pair consistency voting. IEEE transactions on visualization and computer graphics, 25(4), 1693-1706.

Zhou, H., Chen, H., Feng, Y., Wang, Q., Qin, J., Xie, H., ... & Wang, J. (2020). Geometry and learning co-supported normal estimation for unstructured point cloud. In Proceedings of the ieee/cvf conference on computer vision and pattern recognition (pp. 13238-13247).

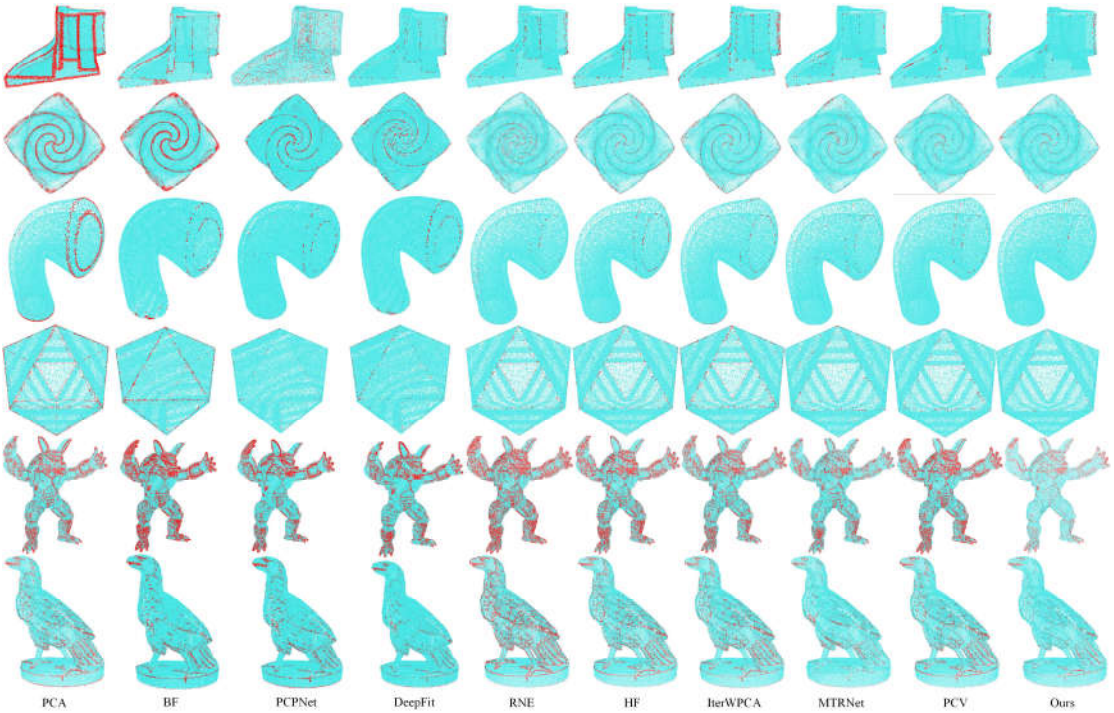


Fig.7

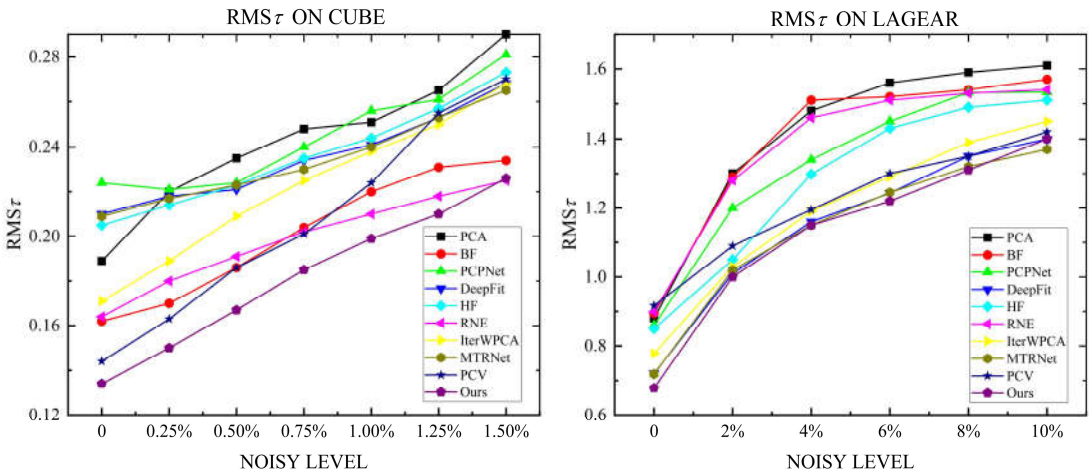


Fig.8

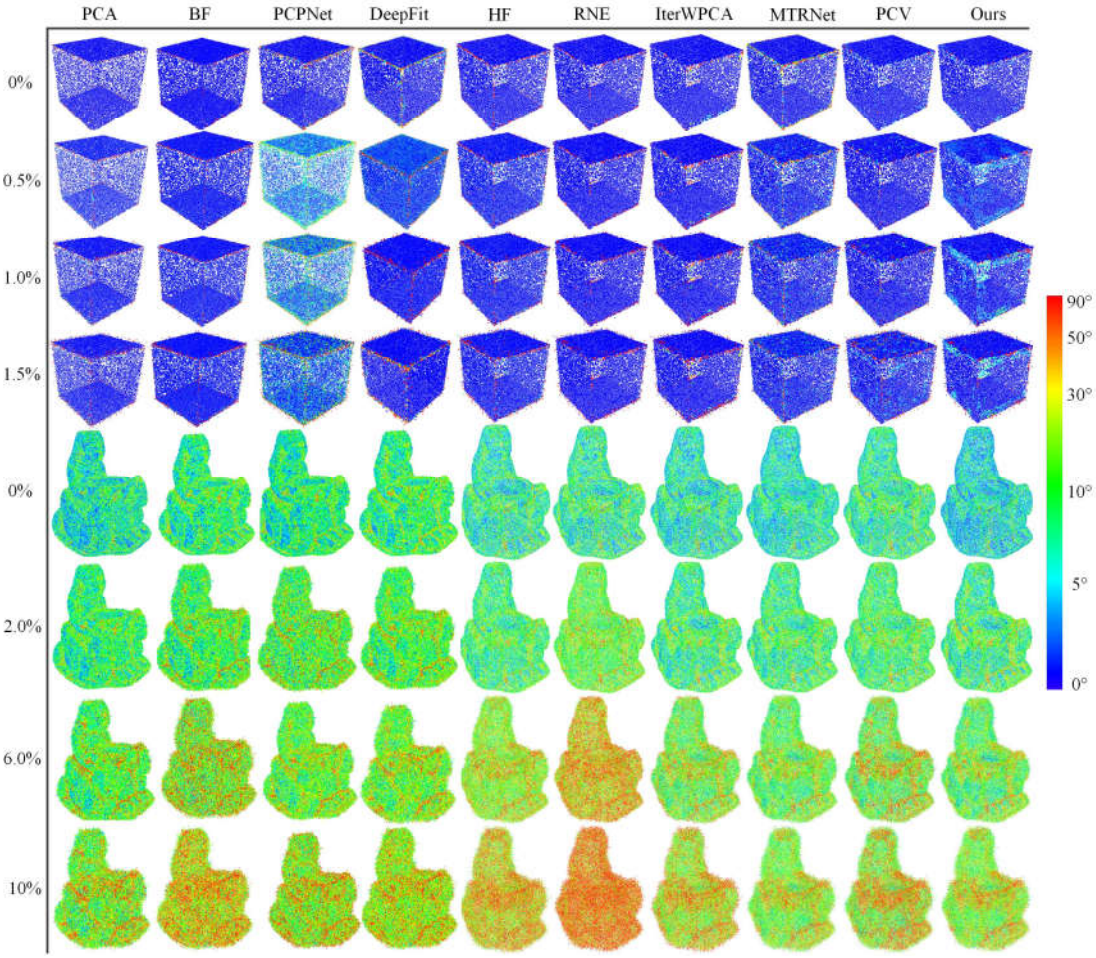


Fig.9

Point 5: Point cloud smoothing is a straightforward application of the presented approach, yet the experiments shown are not very convincing. This method has to be compared against [5] and Alexander Agathos, Philip Azariadis, Sofia Kyratzi, Elliptic Gabriel Taubin smoothing of point

clouds, Computers & Graphics, Volume 106, 2022, Pages 20-32, https://doi.org/10.1016/j.cag.2022.05.009.

Also, the work of Shachar Fleishman, Iddo Drori, Daniel Cohen-Or, Bilateral mesh denoising, ACM Transactions on Graphics, Volume 22, Issue 3, July 2003 pp 950–953, https://doi.org/10.1145/882262.882368

uses a procedure for obtaining normals that is directly applicable to point clouds and can be considered also in this work.

Response 5: Thank the reviewer for the comments. In order to provide the reader with a more comprehensive understanding of the effectiveness of the proposed methods, we have added these three methods as comparative experiments in section 6.2 of the manuscript, as shown in Figure 10, taking into account the research focus of the three articles and the experimental setup within the literature. In the experiments of Section 6.2, we demonstrate the performance of our method in point cloud filtering applications and surface reconstruction results that are more in line with the experimental philosophy of these three papers.
On the other hand, the comparative experimental methods in the literature, such as MLS,LOP,WLOP, RIMLS, etc., are rarely used as comparative experiments in the field of normal estimation that our paper concerned with. Our comparison methods are more likely to be PCA, PCV, HF,PCPNet and other normal estimation methods in the reference.

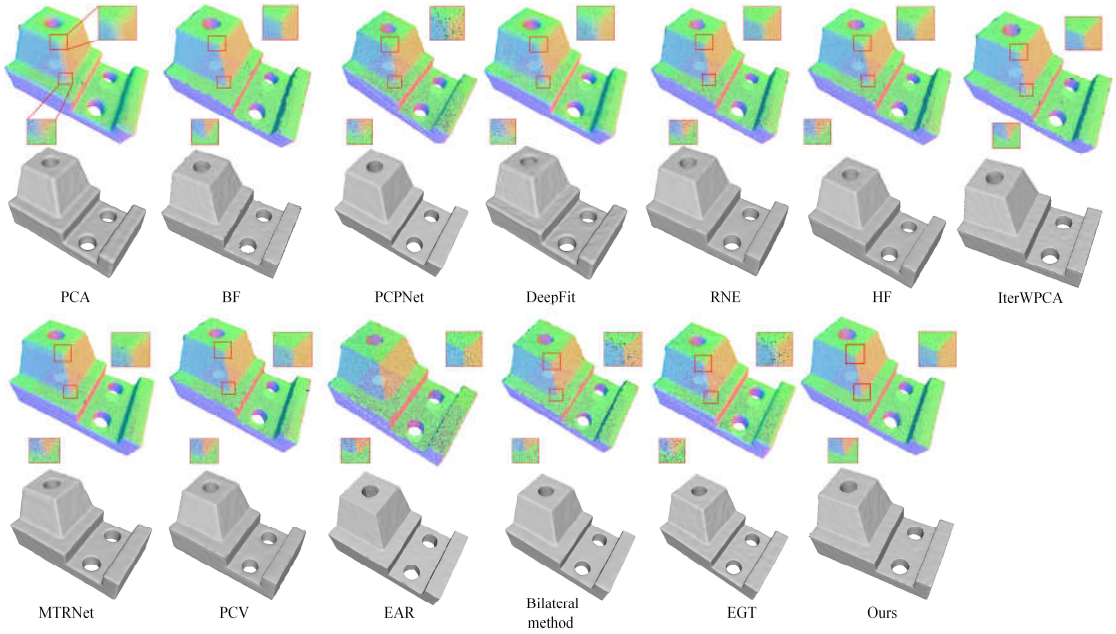


Fig.10

Point 6: Line 255: "robust RANSAC algorithm": what is this algorithm, it is different than RANSAC (which is not performing well with noisy data)?

Response 6: Thank the reviewer for the comments. We've recognized that this description in the

previous original manuscript was not correct. The robust RANSAC algorithm in the original manuscript is the general RANSAC. An advantage of RANSAC is its ability to do robust estimation of the model parameters, i.e., it can estimate the parameters with a high degree of accuracy even when a significant number of outliers are present in the data set. For a more accurate description, We have changed the " robust RANSAC algorithm " to " RANSAC algorithm ".

Point 7: Real-world scan data can easily reach several hundred thousand of points. How your method performs with this amount of data?

Response 7: As you are aware, point cloud processing for real, large-scan data is often computationally intensive. Our experiments on real scenes are designed to verify the effectiveness of the proposed method on a wide range of data, and therefore generalized preprocessing methods for point clouds, such as sampling, filtering, are performed on real-world data. Of course, the latest acceleration techniques can be considered in future work, such as the GPU acceleration methods used in the literature you have given(i.e.,[Alexander Agathos, Philip Azariadis, Sofia Kyratzi, Elliptic Gabriel Taubin smoothing of point clouds]). It should be noted that, based on previous experience, reviewers have asked us to perform validation experiments on both synthetic and real-world data types, so we did this to verify the validity of the method, and the method in this paper is not optimized specifically for large-scene data.

Minor comments:

Point 8: The paper suffers from bad formatting. There are many problems with missing references, equations and symbols floating above the text. There are also some spelling problems.

Line 25: The normal information,
Line 209: via the normal mollification methodology
Line 366: robustness to noise
Line 143, reference for NERL
Algorithms 1, 2, The pseudocode is written with centering, this is not acceptable. Please use left indent and provide the necessary tab spaces to make the code clear.

Response 8: The reviewer's suggestions have been incorporated. It was only after we received the revised manuscript that we noticed many basic problems with the formatting, references, formulas, Algorithm and symbols. We suspect that this may have been caused by the submission system, which did not exist in our original manuscript, and I hope that the current version will satisfy you. We have corrected all the spelling problems you pointed out and would be grateful for your correction.