Kernel Density Filtering for Noisy Point Clouds in One Step

M.A. Brophy, S.S. Beauchemin, J.L. Barron

Department of Computer Science
University of Western Ontario, Canada
{mbrophy5,beau,barron}@csd.uwo.ca

Abstract

We present a method for filtering noisy point clouds, specifically those constructed from merged depth maps as obtained from a range scanner or multiple view stereo (MVS), applying techniques that have previously been used in finding outliers in clustered data, but not in MVS or range scanning. We estimate the probability density function (PDF) over the space of observed points via a technique called kernel density estimation. We utilize Mahalanobis distance and a variable bandwidth for weighting kernels accordingly, based on the nature of neighbouring points. Further, we incorporate a distance metric called the Reachability Distance that, as we show in our results, gives better discrimination than a classical Mahalanobis distance-based metric. With the addition of this nearest neighbour metric, we can produce results that are ready for meshing without any post-processing of the cloud. We mesh our filtered point clouds using a traditional surface fitting technique that is unequipped to deal with noise to demonstrate the efficacy of our method.

1 Introduction

Following a renaissance in energy-based MVS methods in the literature, there has been a return to methods that merge depth maps from multiple views to generate a representative point cloud. This change in methodology can be traced back to the work of [Goesele et al., 2006], whose simple method generated depth maps for each camera using adjacent views, and then merged those depth maps using third-party software designed to merge range images into a complex model. Prior to this, the MVS literature was dominated by methods that evaluated photo-consistency over a dense grid and then used an energy minimization technique to extract a representative surface, which is often the minimal surface.

A cursory glance at recent results on the Middlebury multi-view stereo data sets [Goesele et al., 2006] indicates incredible improvement of modern stereo matching algorithms over their predecessors. Such advancements are possible because of the improvements in disparity map generation like ordering constraints, bi-directional image matching, etc. That said, merging multiple depth maps and fitting a surface to the resultant point cloud remains a challenging endeavour, at least partially because of the presence of outliers and the general location of these outliers. These outliers can be very difficult to filter, as outlier clusters can occur both near to and far away from the true surface. Mahalanobis distance-based density estimation cannot correctly identify points that are close to the center of these outlier clusters as not being part of the surface.

The goal of this paper is to demonstrate that it is possible to fit a surface to a point cloud with very large quantity of outliers (a ratio of 5:1 outliers to inliers), by filtering using anisotropic kernel density estimation with variable bandwidth, and subsequently fitting a surface to this filtered cloud using standard surface meshing software [Cignoni et al., 2008]. We take clean, merged point clouds and populate them with noise. This way, we know the ground truth and can thus quantify this method's ability to discriminate inliers from outliers with the Receiver Operating Characteristic (ROC) curves [Provost and Fawcett, 2001].

Our method uses a process called kernel density estimation to construct a probability density function over the space of discrete data that we obtain from measured data. The target application is filtering point clouds obtained from MVS data. [Xi et al., 2009] and [Schall et al., 2005] utilize a similar process, but our method differs from theirs in that we use a variable bandwidth based on the nearest neighbour of each point that contributes to a density estimate, as inspired by [Latecki et al., 2007, Loftsgaarden and Quesenberry, 1965,

 ISBN 978-0-9934207-0-2

Terrell and Scott, 1992]. These works utilize variable bandwidths in finding outliers in clustered data. Our key observation is that these metrics are also very effective in filtering MVS and range scan data. Mahalanobis distance works well for discrimination of points near the true surface, in our experience better than Euclidean distance-based methods, but is prone to accept false positives near the center of a cluster of outliers. [Xi et al., 2009] note that, when necessary, they apply their filter repeatedly apply their filter until they get a result that is visually acceptable. Our method provides sufficient discriminatory ability such that it can be applied to a noisy point cloud once and the result can be meshed as is. We quantify our results and compare the discriminatory ability of our method with the Mahalanobis distance-based method used in [Xi et al., 2009].

2 Previous Work

[Lu et al., 2005] utilize tensor voting (TV) with a minimal surface-based fitting scheme to reconstruct surfaces from highly noisy (1:1 signal-to-noise ratio) point clouds. They use the level set formulation of [Zhao et al., 2000] to evolve an initial implicit surface to fit the points, but add an extra term to influence the motion in the direction of the tensor.

When working with noisy data, one way to deal with outliers is to average them out. This is the method utilized by [Goesele et al., 2006]. The software they use, *VRIP*, converts each depth map into a signed distance function (SDF), and then merges these SDFs using a weighted averaging based on the angle between the observed points and the sensor in each depth map. The area over which a point can be averaged with another point is referred to as its “ramp". Such a method results in smoothing of observations, and subsequent depth based methods that explicitly filter out a subset of the point cloud construct a surface from actual, unsmoothed observations.

[Li et al., 2010] identify “tracks", matched features that are found in at least three different views of an object, and then use bundle adjustment to recover the 3D point. If the reprojection error is above a certain threshold in one of the images, the point in question is discarded.

[Campbell et al., 2008] keep multiple hypotheses for prospective matches, and then use a spatial consistency measure in a Markov Random Field minimization scheme to recover better matches. If a point's hypotheses are not spatially consistent with its neighbours, it is discarded. [Bradley et al., 2008] construct point clouds using multi-scale matching and then use an iterative filtering method for outlier detection on the resultant point cloud. They compute the projection of a point and its neighbours to a plane and then evaluate the fit using a density function.

[Xi et al., 2009] is the reference point for this work. They merge depth maps constructed from multiple views and use an anisotropic kernel density estimation method combined with a projected line search to obtain the maximum along each normal to find the maximum area of density on each normals path. We forego the use of reprojection error and “ramps" and instead use a density estimator to determine the quality of an observation. [Schall et al., 2005] also use an anisotropic kernel for filtering. Their method is similar to [Xi et al., 2009] in that they use an iterative method to move points along the normal direction to areas of maximal density. Further, they eliminate noise from high quality scans, and generate smooth surfaces from very high quality scans. Our focus is somewhat different, in that we study the circumstance where the quantity of outliers equals or exceeds the number of inliers, testing the ability to discern between inliers and the types of outliers that we see in photometric stereo, i.e. those that are both clustered near and far from the true surface. Further, we wish to simply filter the point cloud, as opposed to iteratively shifting the points to the area of highest density along the normal.

We assert that if a slightly better density estimator is used, one that uses both Mahalanobis distance, an adaptive bandwidth and the reachability distance [Breunig et al., 2000], we can mesh resultant point clouds without any further complication. We use a more advanced nearest neighbour metric, point clouds from range scans and MVS can be filtered without the projected line search while yielding an output cloud that can be meshed using an off-the-shelf method without any other pre-processing.

3 Obtaining a Probability Density Function from Measured Data

The easiest way to construct a probability density function from a set of points is a binning approach, similar to the construction of a histogram. Consider the problem in 1D: Say we have a set of measured data

$X = [x_1, x_2, \ldots, x_n]$ where each data point x_i is a scalar-valued observation. If we construct a set of k bins and simply count the number of items that fall within each bin, we can easily construct a histogram.

A number of questions arise:

- How many bins should we use?
- Where should our bins start and end?
- How does this strategy scale in higher dimensions, i.e., how do we determine the orientation of the bins in multiple dimensions?

The next section introduces a better method for constructing a PDF.

3.1 Kernel Density Functions

Kernel density functions propose to solve the problem of obtaining a probability density function in a different way. Instead of creating arbitrary bins of data, the density is instead evaluated at each point, using the distance to neighbouring points as input to a kernel function, the most commonly used of which is the Gaussian kernel [Xi et al., 2009]

$$K(x) = \frac{1}{(2\pi)^d} \exp\left(-\frac{x}{2}\right), \tag{1}$$

where d is the dimension. This is referred to as the Parzen Window technique [Parzen, 1962]. For each point $x_i \in X$, we rely on all points within a predefined radius to calculate the density of x_i using the kernel $K(x)$

$$f(x_i) = \frac{C_{k,d}}{n \cdot h} \sum_{j=1}^{n} K\left(\frac{||x_i - x_j||^2}{h}\right), \tag{2}$$

where the points x_j are the n neighbouring points of x_i within radius r, $C_{k,d}$ is a weight constant and h is the bandwidth.

Intuitively, $f(x_i)$ will be close to 1 if the sum of distances between x_i and its neighbouring points is small compared to the bandwidth. In other words, areas that contain a large number of points inside of their radius will yield a large density estimate and thus are more likely to be considered inliers than outliers.

3.2 Mahalanobis Distance

[Xi et al., 2009] utilize a more advanced method for filtering, based on the observation that the distribution of noise in point clouds tends to be anisotropic in nature. Thus, they evaluate an anisotropic kernel f of fixed radius r and shape at each point x to estimate its density utilizing its neighbouring points within distance r. Instead of using the L^2 distance between points within r, they find the distance to the center of mass by making use of

$$f(x) = \frac{C_{k,d}}{n} \sum_{i=1}^{n} K(d_\Sigma(x, x_i)), \tag{3}$$

where the kernel K is as defined previously. $d_\Sigma(\cdot,\cdot)$ is the Mahalanobis distance, which is defined as

$$d_\Sigma(x, x_i) = \left((x - x_i)^T H^{-1} (x - x_i)\right)^{1/2},$$

where the covariance matrix

$$H = DD^T,$$

can be constructed using

$$D = (x_1 - x, x_2 - x, \cdots, x_n - x).$$

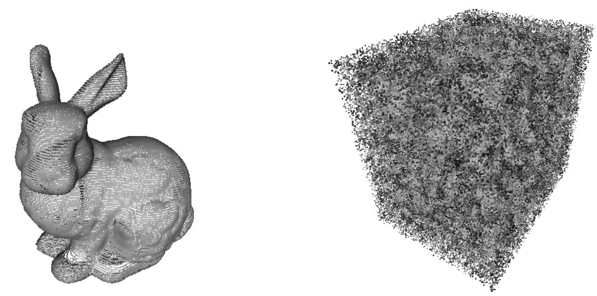
They find the location of highest density within the neighbourhood (r) and then use the distance to this point as the distance for the kernel to evaluate. This method discriminates between inliers and outliers when near the “true" surface much more robustly than the basic kernel density estimation method that we defined in Equation (2). We found that this method still had good discriminatory power when the signal-to-noise ratio was 1:10, i.e. we added 10 randomly generated outliers for each inlier in the point cloud’s bounding volume. Their method is prone to errors when there are clusters of outliers in a small area, a common occurrence.

This method isn’t new, it has been used in outlier estimation when dealing with clusters of data in the past. Likewise, our work is based on well-founded principles that are known in the literature. It hasn’t been applied to this domain, and its discriminatory ability is notable.

3.3 Local Bandwidth Estimation

The idea to use a local estimate of bandwidth $h(x_i)$ comes from [Latecki et al., 2007] who applied it to detecting outliers in clusters of data. It effectively unweights points whose n-th nearest neighbour is any larger than a very small distance from the point itself. Applying this to surface fitting makes a lot of sense, as points that lay near the true surface should have a neighbour in its very near vicinity.

4 Methodology



(a) Bunny. (b) Bunny, noise added.

Figure 1: The *Bunny* point cloud with 5:1 ratio of noise to inliers added.

We attempt to remove outliers via a method that differs from the previously described one in two ways. Based on the density of the nearest neighbour, we can weight each kernel accordingly. In other words, if a neighbouring point x_i itself has a low density, the bandwidth $h(x_i)$ will be lower and thus the contribution to the magnitude of $f(x)$ will be smaller than an equally distant point that exists in an area of higher density.

$$f(x_i) = \frac{1}{n}\sum_{j=1}^{n} \frac{1}{h(x_j)^d} K\left(\frac{d_\Sigma(x_j, x_i)}{h(x_j)}\right) \tag{4}$$

The bandwidth is the distance of the nearest neighbour to x_i, the dimension d is 3, $n = size(NN(x_i)$, where $NN(\cdot)$ is the set containing the nearest neighbours of x_i, the points within the radius r. The Gaussian kernel is as defined previously. The bandwidth,

$$h(x_j) = \min(d_\Sigma(x_j, x_k))$$

where $x_k \in NN(x_j)$, i.e. x_k is the nearest neighbour to x_j, when Mahalanobis distance is used to determine the “closeness" of two points. This method differs from the anisotropic kernel density method described in the previous section in that the density of a point *relies on the density of its neighbouring points*. In other words, we could have a point x_i and its nearest k points, and in the previous method, its kernel density estimate $q(x_i)$ would be the same irrespective of the points surrounding these neighbours.

4.1 Reachability Distance

We can extend this idea further by replacing the numerator of Equation (4) with a more robust metric called the *reachability distance*, where

$$\mathbf{rd}(x_i, x_j) = \max(d_\Sigma(x_j, x_i), d_\Sigma(x_k, x_j)). \tag{5}$$

This yields

$$f(x_i) = \frac{1}{n}\sum_{j=1}^{n} \frac{1}{h(x_j)^d} K\left(\frac{\mathbf{rd}(x_i, x_j)}{h(x_j)}\right), \tag{6}$$

and is thus composed of both the distance from x_i to its neighbours x_j, and distances of neighbouring points x_j to their nearest neighbour (x_k). If $rd(x_i, x_j) = d_\Sigma(x_i, x_j)$ then the inside of K is $-\frac{d_\Sigma(x_i, x_j)}{2d_\Sigma(x_j, x_k)}$ where the

 ISBN 978-0-9934207-0-2

numerator is greater than the denominator, and thus yields an increasingly smaller value as this difference increases when evaluated by the exponential. If $rd(x_i, x_j) = d_\Sigma(x_j, x_k)$ then the inside of the exponential is $-\frac{1}{2}$ and thus gives the minimal value.

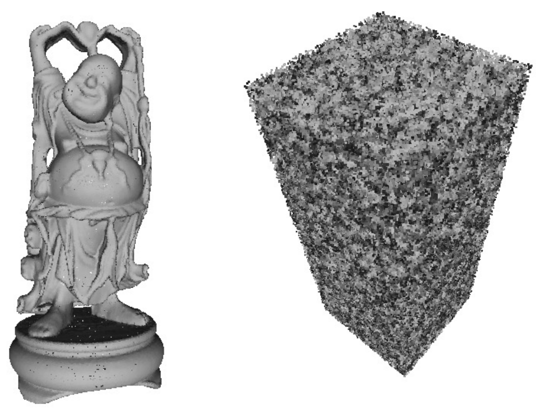
If this neighbour x_j's nearest neighbour is quite far away, i.e., x_k is a somewhat "isolated" point, it gives us very little information about the nature of point x_i, as being in the same neighbourhood as a likely outlier is not a great clue that x_i is an inlier. If, however, the distance to x_j is larger than the distance to x_j's nearest neighbour, we can say that having such a point in the neighbourhood is good evidence, and it is thus more likely that x_i is indeed an inlier!

5 Kernel Density Filtering on Data with Additive Noise

We approach it first using anisotropic kernel density estimation, where we evaluate the density of the data within a set radius of each data point. (3) evaluates the contribution of each point within this area, and takes into account the local density of each of these points as well.

We filter the signal by removing points whose density are below some threshold $\tau \in [0, 1]$. The method struggles with clusters of outliers though, and a substantial percentage of inliers are removed before the outliers fall below τ.

Interestingly, when we use the nearest-neighbour kernel density estimate, even in conjunction with the less discriminative L^2-norm, we still recover a much more accurate signal. That said, the performance of the anisotropic kernel density near the signal is better.



(a) Buddha statue. (b) Statue with noise added.

Figure 2: The *Buddha* point cloud with 5:1 ratio of noise to inliers added.

6 Results

We test our algorithm on the *Bunny* and *Buddha* data sets from Stanford and add random noise of varying quantities to determine the ability of our density-based method to discriminate between inliers and outliers. To quantify the filter's ability to discriminate between an inlier, i.e. a member of the normal class ("NC") and an outlier ("C"), we generate a ROC curve, which plots the detection rate (r_D) versus the false alarm rate (r_{FA})

$$r_D = \frac{TP}{TP+FN} \quad (7)$$

$$r_{FA} = \frac{FP}{FP+TN} \quad (8)$$

where *TP* is the number of true positives, *FN* is the number of false negatives, and *FP* is the number of false positives and *TN* is the number of true negatives. The nature of these terms is explained the confusion matrix seen in Table 1. A perfect ROC curve has an area of one beneath said curve.

ISBN 978-0-9934207-0-2

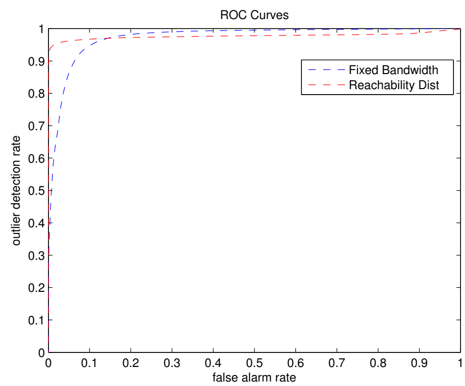


Figure 3: ROC curve for anisotropic filters on the *Bunny* point cloud with 5:1 noise. The reachability distance/adaptive method is in red.

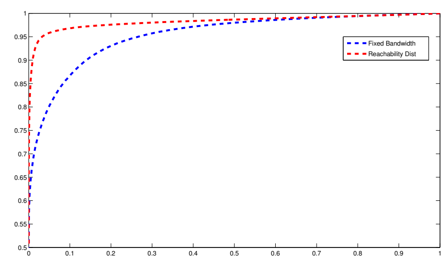


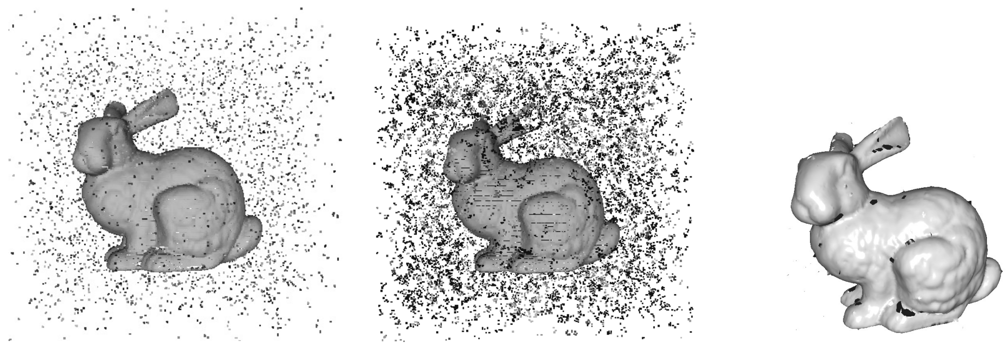
Figure 4: ROC curve for anisotropic filters on the *Buddha* point cloud with 5:1 noise. The reachability distance/adaptive method is in red.

As can be seen from the ROC curves in Figures 4 and 3, our method offers excellent discrimination between inliers and outliers in both circumstances. When one uses a strictly Mahalanobis-based density estimator on the *Bunny* data, a substantial portion of outliers remain. To make this more clear, we display the remaining points in the resultant cloud in Figure 5. There are far too many outliers in Figure 5(b) to allow for the fitting of a surface points, and true surface points are being thresholded as we remove more outliers. In Figure 5(a), a few outliers remain, but they occur in such low densities that they do not interfere with the subsequent surface fitting. We meshed the points from Figure 5(a) in Figure 5(c) with the ball pivoting algorithm [Bernardini et al., 1999].

Likewise, we see that our method works extremely well on the *Buddha* data set. Most impressive is its ability to handle the thin part of the statue above the head. As we see in Figure 6, despite the addition of 5:1 noise, we can still perform an accurate reconstruction of the surface with our method.

We also experimented with using the k nearest neighbours of x_i for estimation of the Mahalanobis distance in the above examples, but we found that the slight increase in discrimination was not worth the added time complexity.

Adjusting the area of support, r, has an effect on the nature of the filtering. If the algorithm is having trouble removing outliers near the surface, it may be useful to *decrease* the radius of the points that contribute to the density estimate. Increasing r will include more points with a larger distance to a point x_i if it is an outlier, but it will do the same for an inlier. The radius should be large enough to contain a sufficient number of points (for our purposes, ≥ 50), but small enough that the density estimates are excessively "smooth". Ideally, the estimation of f for any outlier near the surface will include a large sampling of inliers (i.e. actual surface points) to weight the center of mass correctly, yielding a small distance to points on the surface, and a large distance to outliers. Further, if there are thin areas on the surface, a small r can be useful to ensure that a density estimate at x is only influenced by its neighbours on the surface, not close by points



(a) Reachability distance. (b) Fixed Mahalanobis. (c), Bunny, meshed.

Figure 5: The *Bunny* point cloud with 5:1 noise, filtered. It was meshed with the ball pivoting method.

	Predicted Outlier Class (C)	Predicted Normal Class (NC)
Actual Outliers	True Positive	False Negative
Actual Normal Class	False Positive	True Negative

Table 1: Confusion matrix describing the different classifications of inliers and outliers.

that belong to a different part of the surface.

7 Conclusions and Future Work

We have demonstrated that our filtering method performs well on challenging data sets, even when the point cloud to which we apply our method is corrupted by large amount of noise. In reality, point clouds obtained from MVS or range scanning are not even near as noisy as our two corrupted point clouds. That said, the nature of the noise may be such that noise resides near the true surface, and it will thus be more difficult for the method to decipher whether a point is an inlier or an outlier.

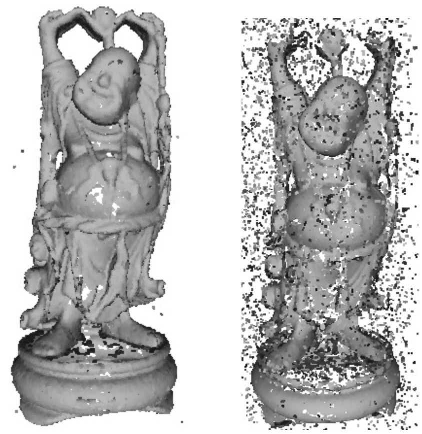
In in future, we would like to automate the process of fitting a surface to our filtered cloud, possibly by including our density estimate in a surface evolution scheme, similar to the level set-based method of [Zhao et al., 2000], with an extra term for density. It might be effective to include the confidence measure of each point in the point cloud from the stereo matching process. In the end, the goal is to obtain extremely accurate multi-view surface reconstructions of objects from multiple views, and a filtering method like the one we've presented is a step in that direction.

References

[Bernardini et al., 1999] Bernardini, F., Mittleman, J., Rushmeier, H., Silva, C., and Taubin, G. (1999). The ball-pivoting algorithm for surface reconstruction. *Visualization and Computer Graphics, IEEE Transactions on*, 5(4):349–359.

[Bradley et al., 2008] Bradley, D., Boubekeur, T., Berlin, T., and Heidrich, W. (2008). Accurate multi-view reconstruction using robust binocular stereo and surface meshing. In *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[Breunig et al., 2000] Breunig, M., Kriegel, H., Ng, R., and Sander, J. (2000). Lof: Identifying density based local outliers. In *Proceedings of the ACM SIGMOD Conference*, Dallas, TX.



(a) Reachability distance. (b) Mahalanobis distance.

Figure 6: Buddha, filtered. The reachability distance-based filtering method leaves almost no outliers.

[Campbell et al., 2008] Campbell, N., Vogiatzis, G., Hernandez, C., and Cipolla, R. (2008). Using multiple hypotheses to improve depth-maps for multi-view stereo. In *Proceedings of the European Conference on Computer Vision ECCV*, volume 1, pages 766–779.

[Cignoni et al., 2008] Cignoni, P., Callieri, M., Corsini, M., Dellepiane, M., Ganovelli, F., and Ranzuglia, G. (2008). Meshlab: an open-source mesh processing tool. In Scarano, V., Chiara, R. D., and Erra, U., editors, *Eurographics Italian Chapter Conference*. Eurographics.

[Goesele et al., 2006] Goesele, M., Curless, B., and Seitz, S. (2006). Multi-view stereo revisited. In *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2402–2409.

[Latecki et al., 2007] Latecki, L., Lazarevic, A., and Pokrajac, D. (2007). Outlier detection with kernel density functions. In *Proceedings of Machine Learning and Data Mining in Pattern Recognition*, pages 61–75.

[Li et al., 2010] Li, J., Li, E., Chen, Y., Xu, L., and Zhang, Y. (2010). Bundled depth-map merging for multi-view stereo. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, 0:2769–2776.

[Loftsgaarden and Quesenberry, 1965] Loftsgaarden, D. O. and Quesenberry, C. P. (1965). A nonparametric estimate of a multi-variate density function. *Annals of Mathematical Statistics*, 36:1049–1051.

[Lu et al., 2005] Lu, H., Zhao, H., Jiang, M., Zhou, S., and Zhou, T. (2005). A surface reconstruction method for highly noisy point clouds. In *Proceedings of Variational, Geometric, and Level Set Methods in Computer Vision, LNCS*. Springer.

[Parzen, 1962] Parzen, E. (1962). On estimation of a probability density function and mode. *Annals of Mathematical Statistics*, 33(3):1065–1076.

[Provost and Fawcett, 2001] Provost, F. and Fawcett, T. (2001). Robust classification for imprecise environments. *Machine Learning*, 42(3):203–231.

[Schall et al., 2005] Schall, O., Belyaev, A., and Seidel, H.-P. (2005). Robust filtering of noisy scattered point data. In Pauly, M. and Zwicker, M., editors, *IEEE/Eurographics Symposium on Point-Based Graphics*, pages 71–77, Stony Brook, New York, USA. Eurographics Association.

[Terrell and Scott, 1992] Terrell, G. R. and Scott, D. W. (1992). Variable kernel density estimation. *The Annals of Statistics*, 20(3):1236–1265.

[Xi et al., 2009] Xi, Y., Duan, Y., and Zhao, H. (2009). A nonparametric approach for noisy point data preprocessing. *International Journal of CAD/CAM*, 9(1):31–36.

[Zhao et al., 2000] Zhao, H., Osher, S., Merriman, B., and Kang, M. (2000). Implicit and non-parametric shape reconstruction from unorganized data using a variational level set method. *Computer Vision and Image Understanding*, 80:295–319.