Eurographics Symposium on Point-Based Graphics (2005)
M. Pauly, M. Zwicker, (Editors)

Robust Filtering of Noisy Scattered Point Data

Oliver Schall Alexander Belyaev Hans-Peter Seidel

Max-Planck-Institut für Informatik, Saarbrücken, Germany

Abstract

In this paper, we develop a method for robust filtering of a noisy set of points sampled from a smooth surface. The main idea of the method consists of using a kernel density estimation technique for point clustering. Specifically, we use a mean-shift based clustering procedure. With every point of the input data we associate a local likelihood measure capturing the probability that a 3D point is located on the sampled surface. The likelihood measure takes into account the normal directions estimated at the scattered points. Our filtering procedure suppresses noise of different amplitudes and allows for an easy detection of outliers which are then automatically removed by simple thresholding. The remaining set of maximum likelihood points delivers an accurate point-based approximation of the surface. We also show that while some established meshing techniques often fail to reconstruct the surface from original noisy point scattered data, they work well in conjunction with our filtering method.

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling

1. Introduction

Point clouds have become increasingly popular in modeling and rendering applications [ABCO*01], [AK04], [PZBG00, RL00, BK03, PKKG03, ZPvBG01] due to improved graphics hardware and technologies for the acquisition of point geometry. Specifically, robust processing of scattered point data is currently a subject of intensive research [Ste99, MVd03, PMG04, FS05, SSB05]. Here robustness means that an estimation / filtering technique works well on noisy data with a small fraction of gross errors ("outliers"). A concise introduction into the field of robust filtering and estimation is available in [PTVF93]. In this paper, we develop a technique for robust filtering of noisy sets of points scattered over surfaces and containing outliers. Such point datasets are routinely generated by optical and photometric range finders. The left image of Figure 1 shows a typical output of a structured light scanner. Beside usual small-amplitude noise the dataset contains many outliers. The right image presents the result of our filtering technique. The outliers are automatically removed and the small-amplitude noise is nicely suppressed.

While low-pass filtering [Lin01], MLS fitting [ABCO*01, MVd04, WPH*04, DGS04] and PDE-based [LP05] approaches remove small-amplitude noise well, eliminating outliers remains mostly a man-

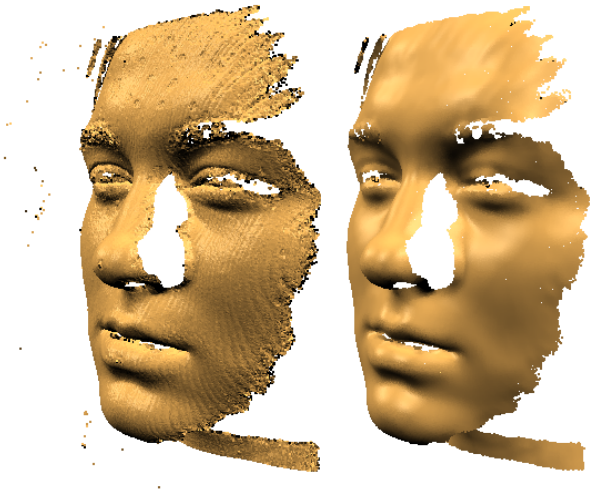

Figure 1: *Filtering of a face scan acquired using a structured light scanner. Initial scattered point data contains small-amplitude noise and outliers (left image). Our method automatically removes the outliers and nicely suppresses small-amplitude noise (right image).*

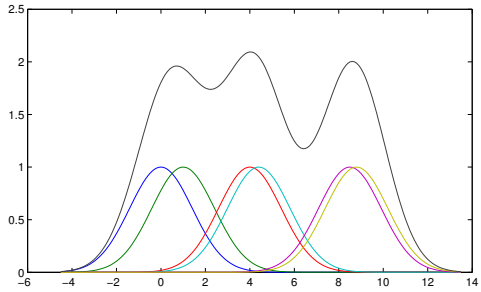

Figure 2: *Illustration of the kernel density estimation technique for 1D scattered point data. Local maxima of the kernel estimation $\hat{f}$ define cluster centers of the original data.*

ual procedure. A concept for the removal of distant outliers can be found in [XMQ04]. Despite of recent progress in extending robust statistics and statistical learning techniques to processing scattered point data [PMG04, ILL*04, SGS05, FS05, SSB05], the problem of automatic outlier identification and removal from scattered point data remains unsolved. We consider our technique as a step towards a satisfactory solution to the problem of automatic robust filtering.

Our method can be considered as a nonparametric kernel density estimation scheme [Ros56, Par62]. Given 3D scattered point data $\mathcal{P} = \{\mathbf{p}_1, \ldots, \mathbf{p}_N\}$, we want to estimate an unknown density function $f(\mathbf{x})$ of the data. A simple kernel estimation $\hat{f}(\mathbf{x})$ of $f(\mathbf{x})$ is given by

$$\hat{f}(\mathbf{x}) = \frac{1}{Nh^3} \sum_{i=1}^{N} \Phi\left(\frac{\mathbf{x}-\mathbf{p}_i}{h}\right). \qquad (1)$$

The smoothing parameter h is called the kernel size and Φ is the kernel function which is usually chosen to be a Gaussian function. Figure 2 illustrates the kernel-based density estimation approach. Local maxima of the kernel estimation $\hat{f}(\mathbf{x})$ naturally define centers of clusters in the scattered data $\mathcal{P}$.

The main idea behind our filtering approach consists of defining a kernel estimation $\hat{f}(\mathbf{x})$ to determine those cluster centers which deliver an accurate approximation of the sampled surface. To detect the local maxima of an appropriately constructed kernel estimation $\hat{f}$ the Mean Shift technique [Che95, CM02, FH75] is used. Then clusters corresponding to the outliers are easily detected by using a simple thresholding scheme.

We also demonstrate that our robust filtering method works nicely in conjunction with mesh reconstruction techniques as Power Crust [ACK01] and Tight Cocone [DG03].

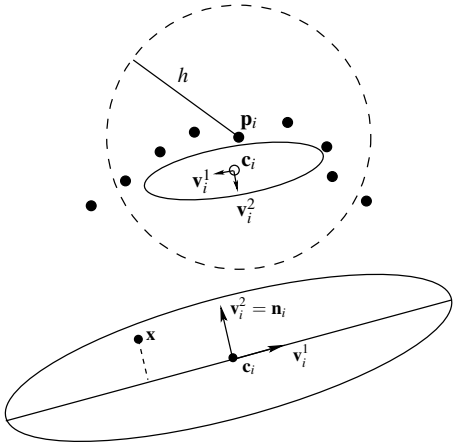

Figure 3: *2D example of the weighted least-squares fitting plane and ellipsoid kernel computation.*

2. Likelihood and Convergence

In this section, we present our statistical method to filter noisy point cloud surface data. We approach this problem by defining a smooth likelihood function L reflecting the probability that a point $\mathbf{x} \in \mathbf{R}^3$ is a point on the surface S sampled by a noisy point cloud $\mathcal{P}$. In order to filter the noisy samples we use an iterative scheme inspired by Mean Shift [Che95, CM02, FH75] to move the points along the gradient of L to maximum likelihood positions.

2.1. Likelihood Function

In order to define the likelihood function L we accumulate local likelihood functions L_i defined for every sample point $\mathbf{p}_i \in \mathcal{P}$. We measure the likelihood $L_i(\mathbf{x})$ for a certain $\mathbf{x}$ considering the squared distance of $\mathbf{x}$ to the least-squares plane fitted to a spatial neighborhood of $\mathbf{p}_i$. Being more specific, we determine the fitting plane by computing the weighted covariance matrix

$$\mathbf{C}_i = \sum_{j=1}^{N} (\mathbf{p}_j - \mathbf{c}_i)(\mathbf{p}_j - \mathbf{c}_i)^T \chi\left(\frac{||\mathbf{p}_j - \mathbf{p}_i||}{h}\right), \qquad (2)$$

where h is the kernel size, χ is a monotonically decreasing weight function and $\mathbf{c}_i$ is the weighted average of all samples inside the kernel. Since $\mathbf{C}_i$ is symmetric and positive semi-definite, its eigenvalues λ_i^l, $l = 1,2,3$, are real-valued and non-negative: $0 \leq \lambda_i^3 \leq \lambda_i^2 \leq \lambda_i^1$. Furthermore, the corresponding eigenvectors $\mathbf{v}_i^l$ form an orthonormal basis. Thus the covariance matrix (2) defines the ellipsoid

$$E_i(\mathbf{x}) = \{\mathbf{x} : (\mathbf{x}-\mathbf{c}_i)^T \mathbf{C}_i^{-1} (\mathbf{x}-\mathbf{c}_i) \leq 1\}, \qquad (3)$$

where the least-squares fitting plane is spanned by the two main principal axes $\mathbf{v}_i^1$ and $\mathbf{v}_i^2$ of E_i and has the normal $\mathbf{v}_i^3 = \mathbf{n}_i$. A 2D example is illustrated in Figure 3. If normals are provided by the scanning device we use them instead of the

estimated normals. Using the squared distance of $\mathbf{x}$ to the least-squares plane we measure the likelihood $L_i(\mathbf{x})$ as

$$L_i(\mathbf{x}) = \Phi_i(\mathbf{x}-\mathbf{c}_i)\left[h^2 - [(\mathbf{x}-\mathbf{c}_i)\cdot\mathbf{n}_i]^2\right]. \quad (4)$$

Thus positions $\mathbf{x}$ closer to the least-squares plane will be assigned a higher probability than positions being more distant. Additionally, we assume that the influence of a point $\mathbf{p}_i$ on the likelihood of a position $\mathbf{x}$ diminishes with increasing distance. To consider this behavior we use monotonically decreasing weighting functions Φ_i to reduce the influence of each L_i. In contrast to radial functions in [PMG04, OBS05] we use a trivariate anisotropic Gaussian function Φ_i which is adapted to the shape of the ellipsoid E_i. This has the advantage that the weighting function is also adapted to the point distribution in a spatial neighborhood of $\mathbf{p}_i$.

To define the likelihood function L modeling the probability that a certain point $\mathbf{x}$ is a point on the sampled surface S, we accumulate the local likelihoods $L_i(\mathbf{x})$ contributed by all points $\mathbf{p}_i$.

$$L(\mathbf{x}) = \sum_{i=1}^{N} w_i L_i(\mathbf{x}) \quad (5)$$

Note that we can easily incorporate scanning confidence measures $w_i \in [0,1]$ associated with each point $\mathbf{p}_i$ by scaling the amplitudes of the likelihood functions. If no scanning confidences are provided we use $w_i = 1$. Figure 4 shows an example of a slice of the likelihood function L.

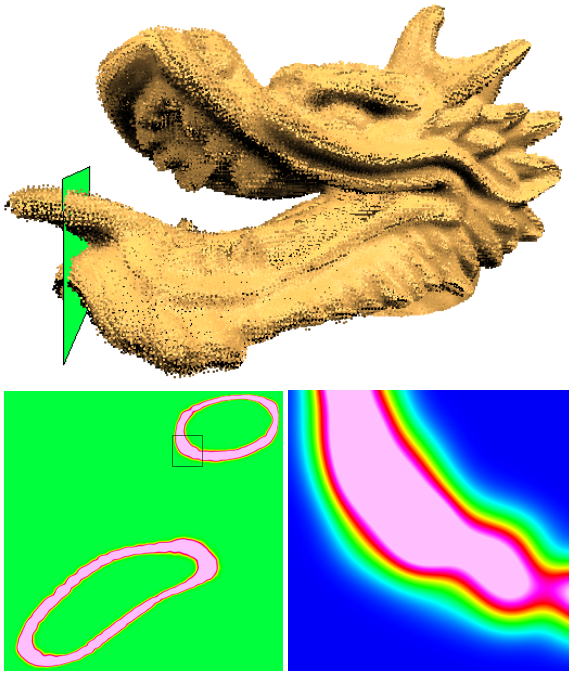

Figure 4: *A slice of the likelihood function L of the noisy Dragon head model (bottom left) and a zoom of the framed region (bottom right). The function values are represented by colors increasing from deep blue to purple. Note that L is a smooth function.*

2.2. Convergence

After determining the likelihood function $L(\mathbf{x})$ we use it to smooth the point cloud by moving all samples to positions of high probability. This means we move the samples to positions which are most likely locations on the sampled surface. To find the local maxima of $L(\mathbf{x})$ we use a procedure similar to a gradient-ascent maximization. We freeze the weighting functions Φ_j since they change slowly and approximate $\nabla L(\mathbf{x})$ by

$$-2\sum_{j=1}^{N} w_j \Phi_j(\mathbf{x}-\mathbf{c}_j)\cdot[(\mathbf{x}-\mathbf{c}_j)\cdot\mathbf{n}_j]\cdot\mathbf{n}_j. \quad (6)$$

To allow a fast convergence of the samples to probability maxima we choose the step-size adaptive as

$$\tau = \frac{1}{2\sum_{j=1}^{N} w_j \Phi_j(\mathbf{x}-\mathbf{c}_j)}. \quad (7)$$

This means that the step size is small near to the probability maximum and increases to the border of each kernel. This provides a fast and stable convergence of all sample points.

Combining equations (6) and (7) we get the resulting iterative scheme

$$\mathbf{p}_i^0 = \mathbf{p}_i \quad , \quad \mathbf{p}_i^{k+1} = \mathbf{p}_i^k - \mathbf{m}_i^k \quad (8)$$

$$\mathbf{m}_i^k = \frac{\sum_{j=1}^{N} w_j \Phi_j\left(\mathbf{p}_i^k-\mathbf{c}_j\right)\cdot\left[(\mathbf{p}_i^k-\mathbf{c}_j)\cdot\mathbf{n}_j\right]\cdot\mathbf{n}_j}{\sum_{j=1}^{N} w_j \Phi_j(\mathbf{p}_i^k-\mathbf{c}_j)}. \quad (9)$$

In order to filter the point cloud $\mathcal{P}$ we apply the iterative scheme individually to every sample. We stop the iterative process if

$$||\mathbf{p}_i^{k+1} - \mathbf{p}_i^k|| < 10^{-4}h. \quad (10)$$

Each sample usually converges in less than 10 iterations.

A feature of our filtering method is the inherent clustering property. As the number of kernels is larger than the number of maxima in the likelihood function L (see Figure 2), several sample points converge to the same probability maximum. We cluster those samples and place one representative point at the local maximum of L. See Table 1 for details on the point reduction rate.

2.3. Adaptive kernel size

So far we only used a fixed radius h to compute the local neighborhoods for the ellipsoidal weight function and least-squares fitting plane computation. However, invariant kernels might not be suitable for datasets with varying sam-

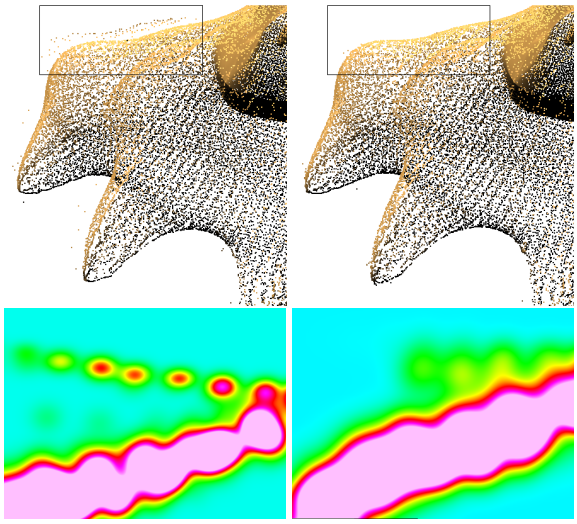

Figure 5: *The effect of adaptive kernels. Left: The Dragon dataset is smoothed using a fixed kernel size. Large amplitude noise at the right foot of the dragon cannot be filtered due to maxima of L distant to the most likely surface. Right: Filtering result of the same dataset using adaptive kernels. Outlying maxima are well damped. Beside very few points, the noisy samples in the rectangular region are filtered completely.*

pling density. To overcome this problem we use the k-neighborhood of each sample $\mathbf{p}_i$ for the PCA analysis to compute the ellipsoidal kernel E_i. In this manner we not only adapt the kernel shape to the point sample distribution in a neighborhood of $\mathbf{p}_i$ but also the kernel size to the spatial sampling density. The motivation behind this choice can be observed in Figure 5. If a fixed radius h is used local maxima of L are created distant to the most likely surface in regions of the point cloud with large-amplitude noise. Those maxima also attract points during the iterative filtering process creating a second layer of points around the most likely surface (left). The usage of adaptive kernels leads to larger kernel sizes in this regions due to the lower sampling density of large amplitude noise. Therefore, kernels of both layers intersect which dampens the effect of local maxima. This results in an improved filtering of large scale noise (right).

3. Results and Applications

This section shows results of our point cloud data filtering technique. We apply our method to real-world datasets from laser and structured light scanners. Furthermore, we present the application of our algorithm to surface reconstruction. In general, surface reconstruction is performed on noisy data reducing the efficiency of surface reconstruction algorithms. We show that the results of well-known surface reconstruction methods can be improved in conjunction with our filtering method.

Dataset	N	M	kernels	filtering	h
Face	180K	114K	1.38s	18.45s	0.8
Sforza (front)	123K	81K	1.33s	21.32s	2
Sforza (side)	143K	95K	1.49s	24.94s	2
Dragon head	485K	170K	23.22s	10m 53s	0.0015
Dragon	2.1M	796K	1m 43s	36m 26s	0.0015
Dragon	2.1M	795K	6m 40s	38m 05s	$k = 250$

Table 1: *Timings for the ellipsoid kernel computation and the filtering for the models presented in this paper. The kernel size h is chosen in the interval of one to ten times the average sampling density of the input data. The character N denotes the number of input samples and M the number of filtered points. The parameter k indicates the number of nearest neighbors used for the adaptive kernel computation. All results were computed on a 2.66 GHz Pentium 4 with 1.5 GB of RAM.*

3.1. Filtering and Outlier Removal

We demonstrate results of our filtering technique in Figure 1 and Figures 5-7. All images are rendered using shaded points. Normals for illustrating the results are computed using PCA analysis with small neighborhoods to avoid blurring effects. Meshes in Figure 7 are displayed using flat shading in order to show faceting. Table 1 summarizes our results and the parameters used to generate them.

Figure 1 shows a point cloud face dataset acquired by a structured light scanner before and after filtering using our method. The raw point cloud suffers from several outliers and ridges which are typical artifacts caused by the structured light. We show this comparison to illustrate the effectiveness of our method for removing outliers and smoothing of difficult datasets. Due to the clustering property of our method, groups of outliers usually converge to a set of single points sparsely distributed around the surface samples. These points can be characterized by a very low spatial sampling density compared to the surface samples. We use this criterion for the detection of outliers and remove them using simple thresholding. In Figure 6 we demonstrate the filtering efficiency of our algorithm on laser scan data. We show this comparison as laser scans are usually affected by different types of noise compared to structured light scans. Due to the different acquisition technique, laser scans are usually not corrupted by ridges and pits caused by structured light, but affected by dense noise. Figure 6 illustrates that high-frequency noise is removed by our method while lower frequency details like hair, mouth and eyes of the Ippolita Sforza Bust are preserved. As noted previously, our method uses adaptive kernels to handle large scale noise. Figure 5 shows that while the dragon scan cannot be accurately filtered using a fixed kernel size, adaptive kernels provide a proper filtering of large amplitude noise.

Figure 6: *Smoothing of two range scans of the Ippolita Sforza Bust. Note that details in hair, mouth and eye regions are accurately preserved.*

3.2. Surface Reconstruction

Surface reconstruction is one of the most fundamental problems in geometry processing. One important group of reconstruction algorithms are Delaunay-based methods. Those techniques are supported by rigorous mathematical results and provide correct reconstructions under specific sampling conditions. Furthermore, they have the great advantage to be able to reconstruct surfaces from points without normals. On the other hand, these methods are sensitive to data with noise and outliers which cannot be avoided in physical acquisition processes. Most Delaunay-based methods are therefore not practical to be applied to raw data. Therefore, making these methods robust for noisy data is currently a field of intensive research.

Our method can be used to filter real-world data before using it for surface reconstruction by computational geometry methods. We apply our filtering technique to noisy point clouds and reconstruct a surface from the preprocessed data. For comparison we also reconstruct a surface from the same noisy point cloud without cleaning it using our method. For surface reconstruction we use two well-known Delaunay-based reconstruction algorithms, namely Power Crust [ACK01] and Tight Cocone [DG03], which are available for scientific purposes. Figure 7 shows results of the comparison. The direct reconstruction of the noisy scattered data does not produce usable results. In contrast, results of both algorithms using the filtered scattered data show significantly improved reconstructions.

4. Conclusion and Future Work

We have introduced a kernel based clustering approach for robust filtering of point cloud surface data. We show that our technique removes noise of different amplitudes and allows for an easy detection of outliers. We demonstrate the effectiveness of our algorithm on real-world datasets acquired using structured light and laser scanners. Furthermore, our method can be used in combination with surface reconstruction algorithms and improves their results on noisy data.

So far our method contains no special treatment of sharp features which are usually present in mechanical models, for instance. A nice work addressing sharp features is introduced in [FCOS05]. We consider this area as an interesting avenue for future work. Furthermore, we plan to augment our filtering technique to out-of-core handling of large datasets. As every point converges independently to a maximum of the likelihood function, parallelization is a straightforward task. This would allow our method to be applied to extremely large models.

Acknowledgments

We thank the anonymous reviewers of this paper for their constructive comments and useful suggestions. We would like to thank Tamal Dey and Nina Amenta for making their surface reconstruction software available and Volker Blanz for the face scan. The Dragon dataset is courtesy of the Stanford 3D scanning repository. The research of the authors was supported in part by the European FP6 NoE grant 506766 (AIM@SHAPE). The Ippolita Sforza Bust is courtesy of the AIM@SHAPE Shape Repository and is provided by IMATI.

References

[ABCO*01] ALEXA M., BEHR J., COHEN-OR D., FLEISHMAN S., SILVA C. T.: Point set surfaces. *IEEE Visualization 2001* (Oct. 2001), 21–28. 1

[ACK01] AMENTA N., CHOI S., KOLLURI R.: The power crust. In *Proceedings of 6th ACM Symposium on Solid Modeling* (2001), pp. 249–260. 2, 5

[AK04] AMENTA N., KIL Y. J.: Defining point-set surfaces. *ACM Transactions on Graphics 23*, 3 (August 2004), 264–270. Proceedings of SIGGRAPH 2004. 1

[BK03] BOTSCH M., KOBBELT L.: High-quality point-based rendering on modern GPUs. In *Proceedings of Pacific Graphics '03* (2003). 1

[Che95] CHENG Y.: Mean shift, mode seeking, and clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence 17* (1995), 790–799. 2

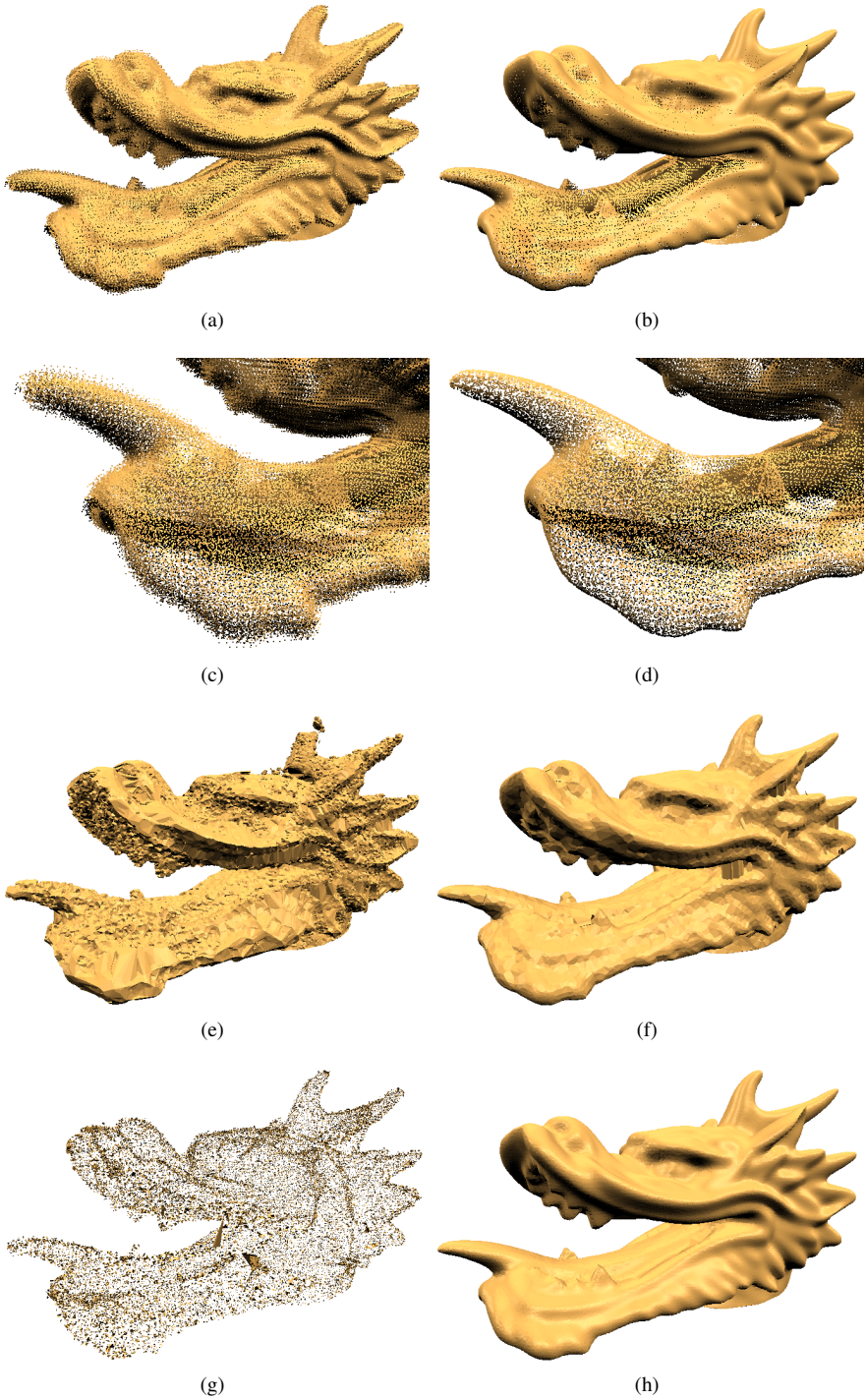

Figure 7: *Figures (a) and (b) present the head of the Dragon scans from the Stanford Scanning Repository before and after our filtering procedure. Figures (c) and (d) show zooms of the images (a) and (b) close to the tongue region. Notice that noise is removed and that the filtered samples indicate a surface. Figures (e) and (g) illustrate Power Crust and Tight Cocone reconstructions from the noisy samples shown in (a). Figures (f) and (h) show reconstruction results from the filtered data shown in (b). While the Power Crust algorithm shows noticeably improved results with small defects (f), the Tight Cocone algorithm reconstructs a smooth mesh (h).*

[CM02] COMANICIU D., MEER P.: Mean Shift: A robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence 24*, 5 (May 2002), 603–619. 2

[DG03] DEY T. K., GOSWAMI S.: Tight cocone: A water-tight surface reconstructor. In *Proc. 8th ACM Sympos. Solid Modeling Applications* (2003), pp. 127–134. 2, 5

[DGS04] DEY T. K., GOSWAMI S., SUN J.: *Smoothing noisy point clouds with Delaunay preprocessing and MLS*. Tech. Rep. OSU-CISRC-3/04-TR17, The Ohio State University, 2004. 1

[FCOS05] FLEISHMAN S., COHEN-OR D., SILVA C. T.: Robust moving least-squares fitting with sharp features. In *Proceedings of ACM SIGGRAPH 2005* (2005). 5

[FH75] FUKUNAGA K., HOSTETLER L. D.: The estimation of the gradient of a density function with applications in pattern recognition. *IEEE Transactions on Information Theory 21* (1975), 32–40. 2

[FS05] FENN M., STEIDL G.: Robust local approximation of scattered data. *Geometric Properties from Incomplete Data* (2005). 1, 2

[ILL*04] IVRISSIMTZIS I., LEE Y., LEE S., JEONG W.-K., SEIDEL H.-P.: Neural mesh ensembles. In *3DPTV 2004* (Los Alamitos, USA, 2004), Aloimonos Y., Taubin G., (Eds.), IEEE. 2

[Lin01] LINSEN L.: *Point cloud representation*. Tech. Rep. 2001-3, Fakultät für Informatik, Universität Karlsruhe, 2001. 1

[LP05] LANGE C., POLTHIER K.: Anisotropic fairing of point sets. *Special Issue of Computer Aided Geometric Design* (2005). To appear. 1

[MVd03] MEDEROS B., VELHO L., DE FIGUEIREDO L. H.: Robust smoothing of noisy point clouds. In *Proc. SIAM Conference on Geometric Design and Computing* (Seattle, USA, 2003), Nashboro Press. 1

[MVd04] MEDEROS B., VELHO L., DE FIGUEIREDO L. H.: Smooth surface reconstruction from noisy clouds. *Journal of the Brazilian Computing Society* (2004). 1

[OBS05] OHTAKE Y., BELYAEV A. G., SEIDEL H.-P.: An integrating approach to meshing scattered point data. In *ACM Symposium on Solid and Physical Modeling* (2005). 3

[Par62] PARZEN E.: On the estimation of a probability density function and the mode. *Annals of Mathematical Statistics 33* (1962), 1065–1076. 2

[PKKG03] PAULY M., KEISER R., KOBBELT L. P., GROSS M.: Shape modeling with point-sampled geometry. *Proceedings of SIGGRAPH 2003 22* (2003), 641–650. 1

[PMG04] PAULY M., MITRA N. J., GUIBAS L. J.: Uncertainty and variability in point cloud surface data. In *Eurographics Symposium on Point-Based Graphics* (Zurich, Switzerland, June 2004), pp. 77–84. 1, 2, 3

[PTVF93] PRESS W. H., TEUKOLSKY S. A., VETTERLING W. T., FLANNERY B. P.: *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press, 1993. 1

[PZBG00] PFISTER H., ZWICKER M., BAAR J. V., GROSS M.: Surfels: Surface elements as rendering primitives. In *Proceedings of ACM SIGGRAPH 2000* (July 2000), pp. 335–342. 1

[RL00] RUSINKIEWICZ S., LEVOY M.: QSplat: a multiresolution point rendering system for large meshes. In *Proceedings of ACM SIGGRAPH 2000* (2000), pp. 343–352. 1

[Ros56] ROSENBLATT M.: Remarks on some nonparametric estimates of a density function. *Annals of Mathematical Statistics 27* (1956), 832–837. 2

[SGS05] SCHÖLKOPF B., GIESEN J., SPALINGER S.: Kernel methods for implicit surface modeling. In *Advances in Neural Information Processing Systems 17*. MIT Press, Cambridge, MA, 2005, pp. 1193–1200. 2

[SSB05] STEINKE F., SCHÖLKOPF B., BLANZ V.: Support vector machines for 3D shape processing. In *Proceedings of EUROGRAPHICS 2005* (2005). 1, 2

[Ste99] STEWART C. V.: Robust parameter estimation in computer vision. *SIAM Review 41* (1999), 513–537. 1

[WPH*04] WEYRICH T., PAULY M., HEINZLE S., KEISER R., SCANDELLA S.: Post-processing of scanned 3D surface data. In *Eurographics Symposium on Point-Based Graphics* (Zurich, Switzerland, June 2004), pp. 85–94. 1

[XMQ04] XIE H., MCDONNELL K. T., QIN H.: Surface reconstruction of noisy and defective data sets. *IEEE Visualization 2004* (2004). 2

[ZPvBG01] ZWICKER M., PFISTER H., VAN BAAR J., GROSS M.: Surface splatting. In *Proceedings of SIGGRAPH 2001* (New York, NY, USA, 2001), ACM Press, pp. 371–378. 1