Evaluation and Optimization of Laser Scan Data

Christian Teutsch* Tobias Isenberg† Erik Trostmann*
Michael Weber*

Abstract

The digitalization of objects from the real world is of great importance, e. g., in many multimedia applications, in 3D computer graphics, and in industrial measurement. In particular, methods of optical measurement are frequently used in this context. This type of 3D scanning yields a point cloud that may contain errors due to system specific characteristics. Considering the example of an optical laser scanner, the data acquisition process is analyzed in order to identify system parameters that can be used for a qualitative evaluation. Several methods for quantitative analysis of the data are discussed that are partially based on a B-spline approximation of points in the point cloud. Furthermore, methods are presented that are particularly well qualified to detect geometrical characteristics such as curvature and edges. These characteristics and the approximated quality of single points are used to reconstruct a revised and optimized point cloud. Finally, the effectiveness of the proposed methods is evaluated based on two exemplary point cloud data sets.

Keywords: reverse engineering, laser scan geometry processing, point cloud analysis, point cloud correction, B-spline approximation

1 Introduction

Geometry models are the very foundation of contemporary three-dimensional computer graphics. In addition to creating new models by using a modeling suite, the use of 3D scanning systems such as laser scanners has recently become more and more common. In this paper we discuss the analysis, evaluation, and correction of the point clouds generated by laser scanning systems.

When reconstructing objects from laser scan data, usually very large data sets have to be processed. Therefore, it is often necessary to minimize the number of points while minimizing the loss of information at the same time. In addition, the generated point cloud usually contains a considerable number of errors. Most of these errors are directly dependent on the measurement system and the scanned object's surface. Outliers and other erroneous points are an important factor when discussing metering precision. Therefore, they have to be detected and removed from the point cloud or corrected in order to get a clean model that can be used as precise measuring data.

*Fraunhofer Institute for Factory Operation and Automation, Department of Intelligent Sensor Systems (ISS), Sandtorstrasse 22, D-39106 Magdeburg, Germany

†Otto-von-Guericke University of Magdeburg, Faculty of Computer Science, Department of Simulation and Graphics, D-39106 Magdeburg, Germany

Finally, an optimization of the laser scanning system can be realized based on the analysis of the data quality of each individual point in the data set. This can be achieved by considering the specific settings of the scanning system, in particular, the parameters of the laser and the camera. Optimizing their positions based on the results of the point cloud analysis results in better point cloud quality in a subsequent scan.

In this paper we present a new method of analyzing and evaluating the point clouds. Based on the features of the laser scan system, a resulting point cloud can be described as a sorted set of B-splines curves. Triangulation algorithms are not needed to calculate normals or to smooth the measured data. Edges can be determined by analyzing the curvature of each point along the B-spline curve.

The correction of the parameters of the scanning system and the optimization of the point clouds leads to a minimum of errors in the data. The major advantage of this procedure lies in that we do not consider and process a possibly erroneous mesh but instead work with the measured data directly—the point cloud.

In the remainder of this paper, we start by discussing related work on the area of 3D surveying and handling the obtained point clouds in Section 2. In Section 3, we analyze the used laser scanning system and its measuring principle in order to obtain potential sources of error. Based on this analysis, Section 4 derives methods for quantifying point clouds from system parameters. The obtained quality values are used in Section 5 to approximate adapted B-spline and NURBS curves in order to allow for optimizing the point cloud by thinning, smoothing, and closing gaps. While some intermediate results are given within these sections, in Section 6 we illustrate final results with a technical element and a plaster model as examples. In Section 7, the proposed methods are summarized and some perspectives for extending our methods in the future are discussed.

2 Related Work

In general, 3D scanning systems produce a set of mostly unsorted 3D sample points. Due to different environmental effects, this set is erroneous and has to be corrected in order to get corrected data for further processing, such as visualization or quality assurance.

The common way to analyze point clouds is to find free-form representations description and use it as the basis for the following algorithms. Popular methods are triangulation algorithms such as *marching cubes* [LC87], *Delaunay triangulation* [Del24], and some shape descriptions based on B-spline and Bézier surfaces. In [Hop94], HOPPE presents a modified version of the original marching cubes algorithm that is valid only for iso-surfaces. By approximating a distance function over portions of the set of 3D data, a triangle mesh is constructed. In the next steps, an energy function is minimized to get a regular and smooth mesh as surface description [HDD+93]. This method leads to smooth approximations but its accuracy cannot be controlled as it is needed, e. g., for precise geometric measurements. For a range scanner, NEUGEBAUER presents a similar approach where the distance function is derived from the range images. A triangulated model is generated from a rough approximation of the object using the marching cubes algorithm. Subsequently, the triangles representing the surface are adaptively subdivided until a pre-defined degree of accuracy has been reached [NK97].

As has been mentioned before, in most cases the scanned data set is erroneous. There are outlying points, gaps, and holes. These errors can lead to misarranged meshes. In [GN98], GROSSKOPF shows the correction of partially incompletely captured surface portions through Geometrical Deformable Models (GDMs). A GDM is defined as a triangle mesh that dynamically deforms by moving vertices based on forces. According to [HDD$^+$93], edges act like springs and a pressure force is defined along the surface normals to cause a smooth deformation. In [Neu96], NEUGEBAUER considers some effects that can appear when working with optical sensors. When scanning a complete surface from different positions, the point cloud consists of several parts that are overlapping. NEUGEBAUER uses redundant areas to connect these partial point clouds and builds a mesh using marching cubes. BOEHLER et al. consider 3D scanning software for point cloud treatment in [BHMS02] with the goal to establish criteria for data cleaning and thinning. On the basis of a description of several problems that can occur during the scanning process (e. g., reflections), they propose some basic techniques that satisfy their criteria.

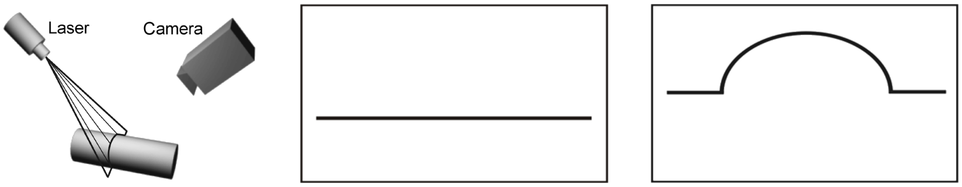
When a freeform surface model has to be generated (e. g., for CAD models), marching cubes is not applicable because only linear triangle meshes are generated. ECK and HOPPE present a procedure for reconstructing a tensor product surface from a set of scanned 3D points [EH96]. They define a surface as a network of B-spline patches and explore adaptive refinement of the patch network in order to satisfy user-specified error tolerances. The main advantage of this method is that single patches represent local surface parts much better than global approximations. Thus, the influence of erroneous parts of a point cloud can be reduced. The goal of the described methods is to compute smooth and esthetically pleasing shapes under some limitations. The practicability of these methods for use in industrial measurement devices and applications depends on high processing speed and accuracy. However, a laser scanner normally produces large and dense point sets with erroneous points. Therefore, the methods described above can only partially be used for effective measurements.

3 Description of the Laser Scan System

For the demonstration of our approach we chose a laser scanning system that operates on the basis of triangulation with *structured lighting*. Structured lighting in this context is the projection of a light pattern (plane, grid, or a more complex shape) at a known angle onto an object. Using this principle, structured lighting can be used to determine the shape of objects in machine vision applications. In addition, it can be used to recognize and locate objects in an environment. These features make structured lighting very useful, for example, in assembly lines for implementing process or quality control. These processes use structured lighting for alignment and inspection purposes. Although other types of light can be used for structured lighting, laser light is usually the best choice because of its precision and reliability. In order to identify parameters of the system that can be used for error analysis and point quality evaluation, we first give an overview of the measurement principles used in the examined laser scanning system.

3.1 Measuring Principle

The projection system of a laser scanner contains several sensors. Each sensor is a combination of at least one laser projection system that projects a light pattern onto the scanned objects and one camera that captures the projected image. The light pattern used most often in optical 3D scanners is generated by fanning out the light beam into a *sheet-of-light* meaning that a single laser beam is expanded by special lenses. When a sheet-of-light intersects an object, a bright line of light is projected onto the surface of the object. Through the topology of the surface this line is distorted. The camera observes this scene from an angle. By analyzing the line of light the observed distortions can be translated into height variations. Therefore, the positions of the laser and the camera are needed. They are derived from the system calibration. For each vertex in the 2D images, 3D point coordinates are computed using the parameters from the calibration. Hence, structured lighting is sometimes described as *active triangulation*. Through a process of rotating the object the whole surface can be mapped into 3D coordinates. The resulting point cloud consists of a set of single lines, each line representing the results from a single sheet-of-light measuring.



(a) Overview of the setup.

(b) Laser line as “seen” by the laser.

(c) Laser line as seen by the camera.

Figure 1: Scanning 3D geometry using the structured laser lighting principle.

The scanning system consists of several parts for hardware and software processing. In the preliminary 2D image, a processing step detects and analyzes the projected laser line. Due to the pixel raster in a digital image, aliasing effects may occur. These effects lead to a high frequent noise in the resulting 3D-point cloud. Therefore, each measured scanline is usually smoothed to reduce these effects. In addition, each measure contains noise and other artifacts such as gaps and holes due to shadowing effects.

It is important to determine or estimate the uncertainty of the measurement for being able to develop algorithms that can guarantee a certain accuracy. The first step is to specify and determine the system’s parameter. In the examined laser scanning system, the following parameters could be identified. Due to the system architecture and the measuring process, the scans result in a sorted set of scans, scanlines, and sublines. The location of the projection system is known, i. e., the positions of the laser and the camera. Since the camera observes the object it is possible to estimate the surface normal of each surface point depending on neighboring points. One parameter that also influences the system is the locomotor system. This moves the object relatively to the sensors and has a certain error tolerance.

3.2 Characterization of Errors

There are many sources for erroneous scan data. The most severe cases will be discussed below. As can be deduced from numerous example scans, most of the outlier and other erroneous points are caused by reflections. In these cases, the high energy laser beam is reflected from mirroring surfaces such as metal or glass. Therefore, too much light hits the sensor of the camera and so-called *blooming effects*[1] occur. In other cases, a direct reflection may miss the camera. In addition, a part of the object may lie in the path from the projected laser line to the camera causing a *shadowing effect*. All these effects are responsible for gaps and holes. At sharp edges of some objects, partial reflections appear. In addition, craggy surfaces cause multiple reflections and, therefore, indefinite point correlations. Other problems are caused by possible range differences originating from systematic range errors resulting from different reflectivity of the surfaces elements. Since the scanner systems are typically used in industrial environments, some atmospheric effects (e. g., dust) may affect the quality of the image obtained by the camera. Furthermore, aliasing effects in the 2D image processing lead to high frequent noise in the generated 3D data.

Therefore, the resulting 3D point data is noisy and partially erroneous. However, a lot of these errors can be minimized by an optimal alignment of the projection system and the object surface so that as few as possible reflections can appear. In order to arrange the setup properly, the quality of the generated point cloud has to be analyzed and evaluated.

4 Point Cloud Quantification

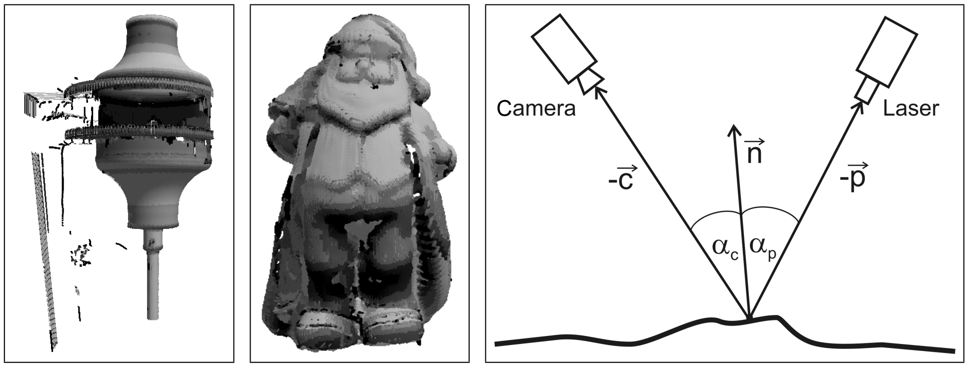
The first step for optimizing a point cloud is to optimize the projection and viewing conditions. Therefore, the quality of the point cloud has to be quantified with respect to the position of laser and camera. Improving the recording conditions leads to less erroneous 3D points. Afterwards, the remaining errors can be detected and evaluated much easier.

4.1 Quantification using 3D Data

Because the laser scan process uses optical sensors, the quality directly depends on the viewing and projection properties. The smaller the angle between surface normal and direction of projection or viewing, α_p and α_c respectively, the better the surface was seen (see Figure 2). In addition, the triangulation between projection vector $\overrightarrow{p}$ and the camera viewing vector $\overrightarrow{c}$ is optimal when the angle $(\alpha_p + \alpha_c)$ defined by them is 90°. With this constraint there are no intersecting errors and no numerical errors when computing the location of the surface point. The more the angle $(\alpha_p + \alpha_c)$ deviates from 90° the worse are the viewing conditions of the surface point.

For the calculation of the point quality, the surface normals have to be known. In general, they can be derived from a triangle mesh. However, for our approach a mesh is not needed. Over a defined neighborhood for each point, a plane is approximated and its normal is used as the normal for the considered point. The neighborhood can be determined very fast since

[1] If too much light hits the sensor of a CCD camera, some cells cannot hold the generated energy and it is distributed to neighboring cells. This is called a blooming effect.



(a) Converter.
(b) Santa Clause.
(c) Projection angles.

Figure 2: Quality of measurements depending on projection angles (bad areas are displayed darker).

the sorting of the scanlines is a known system parameter. To guarantee a consistent normal orientation, the camera position is used as well. Obviously, the surface normal has to point towards the camera that recorded this point. Therefore, the correct surface normal orientation can be computed by calculating the angle between normal vector $\overrightarrow{n}$ and viewing vector $\overrightarrow{c}$. The normal is oriented correctly if this angle is less or equal 90°.

Hence, an important step to optimize a laser scan system is to perform an initial scan and quantify the result based on the position and alignment of the projection system. Depending on the result, the viewing conditions of the object are optimized. In addition, for each point a corresponding quality for the viewing condition can be computed and used for further error compensation. Besides the quality in 3D, each surface point can additionally be evaluated by quantifying its quality in the 2D image processing.

4.2 Quantification Using 2D Data

In addition to the previously discussed viewing conditions, there are factors in the image processing step that may influence the resulting point cloud. These factors are influenced by environmental effects such as lighting, laser light energy, surface type, and camera resolution. For example, a high contrast of a laser line in the recorded image results in a high stability and precise detection. In addition, the line thickness is an important parameter. Thick lines make it harder to find its exact middle. Therefore, the line profile at each vertex has to be considered in the point quantification as well. The higher the slope of an edge orthogonal to the line the better its middle can be detected as a maximum in the profile.

4.3 Additional Quantification

In our experiments we found that erroneous sublines have specific characteristics. On the one hand, they are very short with many high curvature points. High curvatures can be detected by analyzing the approximated curves. On the other hand, sublines can be caused by reflections. These lines are projected somewhere in space and, therefore, have no neighboring scanlines. These aspects can also be used for quantification.

5 Automated Point Cloud Correction

After having discussed methods for quantifying point clouds, we now describe the general approach for correcting single measurements of a point cloud. Measurements and methods of visualization such as triangulation do not require hundreds of thousands or even millions of points because large sets of points need a lot of computation time and memory. Precise measurements, on the other hand, need clean data that is free of noise which may be caused by external effects during a measurement. Therefore, the goal is to minimize the number of points and clean the point clouds from noise. For this purpose, each subline of one discontinuous measured scanline is processed by an approximated B-spline curve.

5.1 B-Spline Approximation

For analyzing a set of points on a curve, B-splines are very helpful. They can be used to obtain a mathematical description from which features can be easily extracted. In addition, B-splines can be used to close small gaps between neighboring sublines. Interpolating these gaps keeps the precision of measurements if the distance between the corresponding sublines is less than a certain threshold (e. g., 1 cm). Otherwise, the interpolated spline segment would follow the assumed geometry insufficiently.
A B-spline curve $x(t)$ of order k (degree $k-1$) is defined over an ordered knot vector T as vectorial polynomial:

$$x(t) = \sum_{i=0}^{n} \mathbf{d}_i N_{i,k,T}(t), \qquad t \in [t_{k-1}, t_{n+1}] \tag{1}$$

with the basis functions $N_{i,k,T}(t)$ and the control points $\mathbf{d}_i$ [PT95]. These B-spline curves offer C^{k-2} continuity at the segment transitions.
To ensure that a B-spline curve approximates given points as well as possible, control points have to be generated from the measured points. Therefore, the distance between the points on the curve X_i and the measured points M_i has to be minimized. A well known method to achieve this is to minimize the quadratic (Euclidean) distance:

$$\sum_{i=1}^{n} \|X_i - M_i\|^2 = min \quad . \tag{2}$$

One way to solve this problem is to use the least-squares-method. Control points D are computed depending on the values of the basis functions N and the measured points M:

$$\mathbf{D} = ((\mathbf{N}^T \cdot \mathbf{N})^{-1} \cdot \mathbf{N}^T) \cdot \mathbf{M} \quad . \tag{3}$$

An approach for solving this problem using different distance norms as a linear programming problem is presented by HEIDRICH et al. in [HBL96].

5.2 Manipulating of the B-spline Representations of Sublines

In the following, we present some approaches to manipulate the B-spline approximated sublines to achieve regular and smooth point clouds.

For further processing, it is useful to generate regularly spaced points. Therefore, the values of the parameter t of the B-spline curve have to be adapted. This can be achieved by choosing a knot vector with parameter distances that are proportional to that of the control points (chord distances) using the following ratio:

$$\frac{t_{i+1} - t_i}{t_{i+2} - t_{i+1}} = \frac{\|d_{i+1} - d_i\|}{\|d_{i+2} - d_{i+1}\|} \ . \tag{4}$$

This approach is also useful to vary the point density on the approximated curve. For example, by calculating 10 regularly spaced points on a curve with an arc length of 10 mm a point density of 1 mm is achieved. For performance reasons, the arc length is estimated as the sum of all distances between neighboring points.

Through aliasing effects in the 2D image processing step, high frequent noise was generated. Therefore, the data has to be smoothed using, e. g., the approximated B-spline curves. HADENFELD presents in [Had98] an iterative method for smoothing B-splines. This method is based on minimizing the bending energy of a thin, elastic bar with a constant cross-section. The corresponding term for a B-spline $x(t)$ is as follows:

$$E = \int (\ddot{x}(t))^2 \quad dt = min. \tag{5}$$

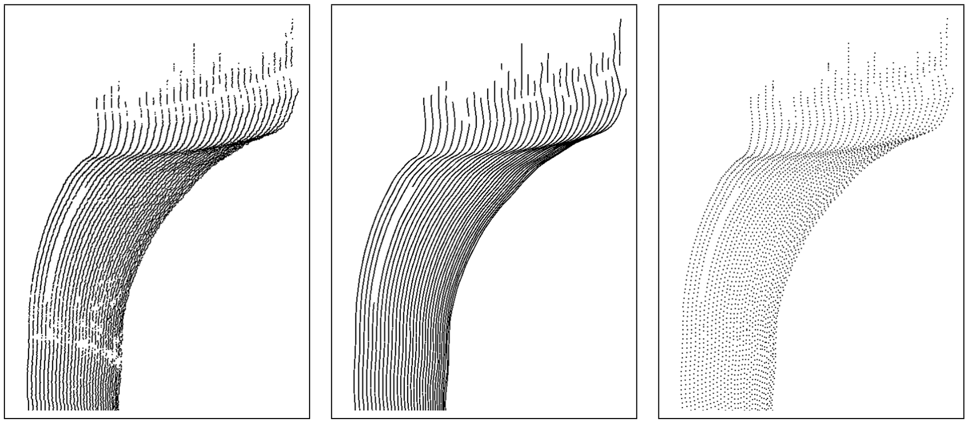
HADENFELD proposes an iterative process of how the control points d_i of a B-spline have to be moved to obtain a smooth curve. Each considered control point $\tilde{d}_i$ is manipulated depending on the control points $\bar{d}_i$ from the last iteration:

$$\tilde{d}_i = -\frac{1}{16}\bar{d}_{i-3} + \frac{9}{16}\bar{d}_{i-1} + \frac{9}{16}\bar{d}_{i+1} - \frac{1}{16}\bar{d}_{i+3} \tag{6}$$

Furthermore, a tolerance can be added to limit the movement of control points with respect to the metering precision. The results of smoothing and thinning an exemplary point cloud are illustrated in Figure 3.

5.3 Detection and Deletion of Erroneous Points

A straightforward approach for minimizing the number of points and correcting the point cloud is to test for the focus space. All parts of a scanned point cloud that lie not within this area obviously should not be used, therefore, they can be deleted. This can be problematic if parts of a subline lie within this area and other parts do not. However, we found that such sublines should not be divided and the potential error can be accepted. In addition, a simple test can be used for sublines that consist of only a few points. In our test, sublines that had no neighbors and consisted of less than 10 points could safely be assumed to be errors.



(a) Original point cloud segment.
(b) Smoothed B-spline approximation.
(c) Smoothed and thinned segment.

Figure 3: Steps for smoothing and thinning.

5.4 Analyzing Curvature Patterns

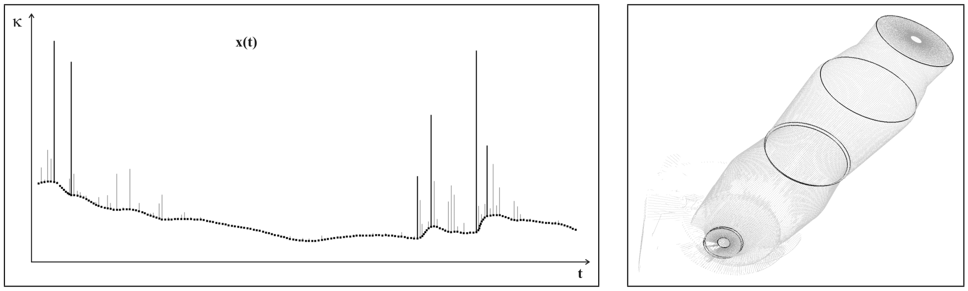
Most of the erroneous sublines exhibit conspicuous curvature patterns. For example, within short distances lots of turnarounds and sharp edges occur. These are indicated by many points with high curvature. In general, the curvature of parameterized curves can be calculated using:

$$\kappa = \frac{\|\overrightarrow{r}'(t) \times \overrightarrow{r}''(t)\|}{\|\overrightarrow{r}'(t)\|^3} \; . \tag{7}$$

In our analysis we found that curvatures of $\kappa > 0.2$ indicate sharp edges reliable. These values can be used to detect potentially erroneous sublines. In many tests with different kinds of objects and curves it was found that sublines with at least 40% of their points having a curvature $\kappa > 0.2$ can considered to be errors and can be deleted. The mentioned parameters are stable in the range of $\pm\, 10\%$.

Sharp edges partially describe the shape of an object. Thus, in addition to use this information for error detection, it can also be very helpful for automatic object recognition (see Figure 4(b)). Further triangulation algorithms could use this edge information for an improved triangle mesh.

However, the curvature can also be used to control the interpolated geometry. While B-splines smooth sharp edges, the curvature can be used to correct the form of the curve at the edges. In addition, the weights from the quantification of the 2D and 3D data can be used to manipulate the path of the curve depending on the quality of each vertex in the image and point in the point cloud. Therefore, rational curves are needed for weighting single points.



(a) Curvatures on a B-spline curve.

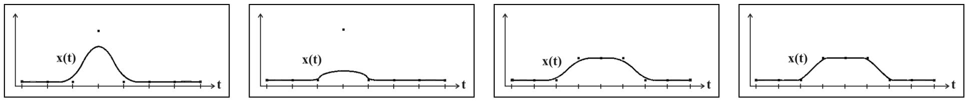
(b) Derived object parts.

Figure 4: Identification of object structures using curvature (e. g., circles, circular parts); $\kappa > 0.2$ in black.

5.5 Reconstruction Using NURBS Curves

The calculation of NURBS curves is similar to that of B-splines. In addition, for each control point d_i a weight h_i can be specified (see Equation 8). This weight determines the influence of the control point on the curve's path. It can be computed based on each point's quality (see Figure 5). Additional descriptions can be found, e. g., in [PT95, Rog01].

$$x(t) = \frac{\sum_{i=0}^{n} \mathbf{d}_i h_i N_{i,k,T}(t)}{\sum_{i=0}^{n} h_i N_{i,k,T}(t)}, \qquad t \in [t_{k-1}, t_{n+1}] \tag{8}$$



(a) Original.

(b) Weighted.

(c) Original.

(d) Weighted.

Figure 5: Two examples for correction of the point on a scanline using NURBS.

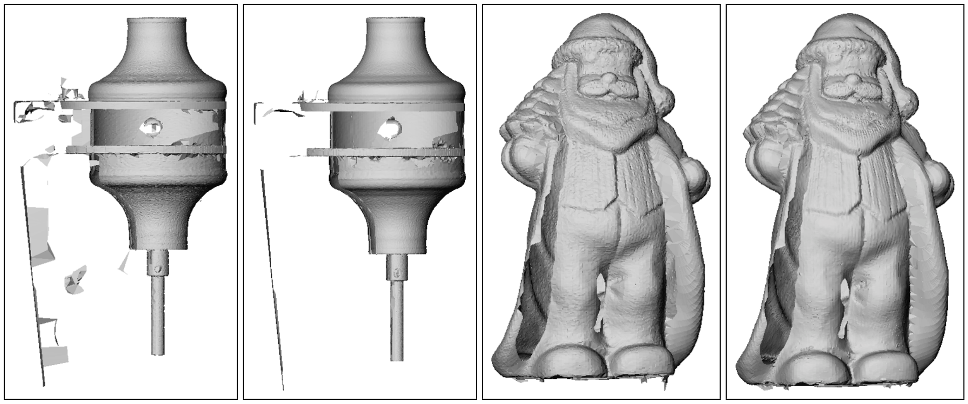
To reconstruct an optimized scanline and, thus, an optimized point cloud, we propose the following rules to weight the control points depending on the curvature properties and the normalized quality values from the quantification (see Section 4):

$$W = \begin{cases} 1 & \text{for all not rateable points} \\ 1 + w_c + w_l + 0.5w_t + 1 & \text{for } \kappa > 0,2 \\ 1 + w_c + w_l + 0.5w_t + \kappa & \text{otherwise} \end{cases} \quad . \tag{9}$$

w_c - weight for the camera viewing angle. w_l - weight for the laser projection angle.
w_t - weight for the triangulation angle (weighted lower due to its angular relationship to w_l and w_c).
κ - curvature.

6 Case Study

The methods described in this paper were tested on a variety of point clouds. For demonstration, we chose a technical element (converter for vehicles) and a plaster model of Santa Clause. The metal surface of the converter caused a lot of errors that could largely be detected and corrected (see Figures 6(a) and 6(b)). In particular, the originally craggy surface of the converter could be smoothed by taking the estimated metering precision of the whole system into account. Considering the model of Santa Clause, the plaster surface caused only few artifacts that largely could be removed. However, the noise in the area of his face could be notably reduced (see Figures 6(c) and 6(d)). The proposed methods are stable, the mostly correct surfaces are not changed substantially. In contrast, the modifications applied to reduce the contained errors are all within the error tolerance of the laser scanning system. In addition, the point density of the tested models could be reduced by up to 75% without appreciable loss of information.



(a) Original. (b) Processed. (c) Original. (d) Processed.

Figure 6: Two examples for triangulated point clouds, before and after processing.

7 Summary and Future Work

In this paper we presented methods to evaluate, quantify, and correct point clouds generated by an optical 3D scanning system automated. We proposed techniques based on the system's parameters in 2D (e. g., contrast and line thickness) and in 3D (e. g., camera and laser positions, focus area, etc.) to estimate the quality of each single point. Furthermore, we approximated point clouds by a sorted set of B-spline curves for iterative smoothing and closing gaps. We derived edge information from these curves and reconstructed scanlines by using NURBS curves with respect to quality and curvature of each single point on it.

The described methods and the acquired systematic coherences allow further considerations of how to use the obtained parameters. The quality values from the point cloud quantification can be used for an automated object and system device positioning to improve the general scanning properties. Triangulation algorithms can use the smoothed and optimized point set as well as the edge information and the computed surface normals for more robust meshing. In addition, the edge information can be used for an automated object recognition. Finally, the developed algorithms can be extended to NURBS and B-spline surfaces to allow interrelated considerations on neighboring scanlines.

References

[BHMS02] W. Boehler, G. Heinz, A. Marbs, and M. Siebold. 3D Scanning Software: An Introduction. In *Proc. of Int. Workshop on Scanning for Cultural Heritage Recording (Corfu, Greece, September 1–2, 2002)*, pages 42–47. International Society for Photogrammetry and Remote Sensing, 2002.

[Del24] B. N. Delaunay. Sur la sphere vide. In *Proceedings of the International Congress of Mathematicians, (Toronto, Canada, August 11–16, 1924)*, pages 695–700. University of Toronto Press (1928), 1924.

[EH96] M. Eck and H. Hoppe. Automatic Reconstruction of B-spline Surfaces of Arbitrary Topological Type. *Computer Graphics*, 30(Annual Conference Series):325–334, August 1996.

[GN98] S. Großkopf and P. J. Neugebauer. Fitting Geometrical Deformable Models to Registered Range Images. In R. Koch and L. J. Van Gool, editors, *3D Structure from Multiple Images of Large-Scale Environments, European Workshop, SMILE'98, Freiburg, Germany, June 6–7, 1998*, volume 1506 of *Lecture Notes in Computer Science*, pages 266–274, Berlin, 1998. Springer.

[Had98] J. Hadenfeld. *Iteratives Glätten von B-Spline Kurven und B-Spline Flächen*. PhD thesis, Technische Universität Darmstadt, 1998.

[HBL96] W. Heidrich, R. Bartels, and G. Labahn. Fitting Uncertain Data with NURBS. In *Proc. 3rd Int. Conf. on Curves and Surfaces in Geometric Design*, pages 1–8, Nashville, 1996. Vanderbilt University Press.

[HDD$^+$93] H. Hoppe, T. DeRose, T. Duchamp, J. McDondald, and W. Stuetzle. Mesh Optimization. In *Proc. SIGGRAPH 93*, pages 19–26, New York, 1993. ACM Press.

[Hop94] H. Hoppe. *Surface Reconstruction from Unorganized Points*. PhD thesis, University of Washington, 1994.

[LC87] W. E. Lorensen and H. E. Cline. Marching Cubes: A High Resolution 3D Surface Construction Algorithm. *Computer Graphics*, 21(3):163–169, July 1987.

[Neu96] P. J. Neugebauer. Scanning and Reconstruction of Work Pieces from Range Images. In *Second IFIP 5.10 Workshop on Virtual Prototyping (Arlington, Texas, May 6–8, 1996)*. Automation and Robotics Research Institute, 1996.

[NK97] P. J. Neugebauer and K. Klein. Adaptive Triangulation of Objects Reconstructed from Multiple Range Imgages. In *Proc. of IEEE Visualization 97*, pages 41–44. ACM Press, 1997.

[PT95] L. Piegl and W. Tiller. *The NURBS Book*. Monographs in Visual Communication. Springer, Berlin, 1995.

[Rog01] D. F. Rogers. *An Introduction to NURBS*. Academic Press, San Diego, 2001.