RECONSTRUCTION OF REAL-WORLD OBJECTS VIA SIMULTANEOUS REGISTRATION AND ROBUST COMBINATION OF MULTIPLE RANGE IMAGES

PETER J. NEUGEBAUER

*Fraunhofer Institute for Computer Graphics
Rundeturmstr. 6, 64283 Darmstadt, Germany
neugeb@igd.fhg.de, <http://www.igd.fhg.de/www/igd-a7>*

Range finder devices are quite useful to measure the geometry of real-world objects. However, the reconstruction of large and complicated objects often requires the acquisition of multiple range images showing different views of the object. With our reconstruction method, the relative orientations of all range images are determined via simultaneous registration of all range images. The registration process is based on a least-squares approach where a distance metric between the overlapping range images is minimized. Registration errors are not accumulated, and it is even possible to reconstruct large objects from an arbitrary number of small range images. A resolution hierarchy accelerates the registration substantially. In order to exploit redundancies, the overlapping surfaces are optimally adapted to each other. The processed range images are converted to a signed distance function, which is based on the idea that no part of the object can lie between the scanner and the measured surface in any view. According to the signed distance function an intermediate volumetric model is sculptured out and polygonalized with isosurface techniques. The accuracy of the generated mesh is improved by moving its vertices onto the surface implicitly defined by the registered range images.

Keywords: range images, reconstruction, registration, sculpturing, implicit surfaces, polygonalization

1. Introduction

3D sensors providing range images are able to scan the shape of an object from one view. However, it is not possible to scan the complete object at once due to topological and geometrical limitations. First, parts of the object are occluded or may lie in shadow. Second, the object might be significantly larger than the 3d sensor is able to capture in one view. In order to simplify the scanning procedure, we must assume that the relative orientations between the different 3d sensor views are unknown. Thus, several unregistered range images showing only partial views of the object must be registered in order to calculate the unknown relative orientations.

The reconstruction of objects is divided in the steps registration of all range images and integration of all range images in one complete geometrical model. The registration is a highly nonlinear problem and hence an initial estimation of the rel-

ative orientation is required. Interactively selected point correspondences are used in our approach to compute the initial estimate. For the reconstruction of large objects from several small range images, it is very important that the registration errors are not accumulated. The unique feature of our approach is that we solve this problem by a simultaneous registration of all range images. Normally, the simultaneous registration of numerous range images requires a great deal of computation time. With our resolution hierarchy, the computation time of the iterative registration is significantly reduced. With our statistical termination criterion, the registration is iterated until all information pertaining to the actual resolution is exploited, and then we switch to the next finer resolution. Thus only few iterations are necessary in the highest resolution.

In order to exploit redundancies and to enhance the range image quality, we present a technique to optimally adapt the overlapping surfaces to each other. Through each measured point a ray is casted along the viewing direction of the 3d sensor. The ray intersects with all overlapping surfaces and the intersection points are robustly combined in order to find a correction for the original measurements.

In section 8 the range images are converted to a signed distance function. Every point in space lying between the surface and the 3d sensor yields a positive distance value. This works quite well even in case of self-occlusions. In order to uniquely determine the object the signed distance function is converted to an intermediate volumetric model. Since the volumetric model may contain different connected segments the segment which most probably represents the object is selected. All other segments are deleted. The volume describes the topology of the object and approximates its shape within the desired level of detail. By finding isosurfaces in the volumetric model a polygonal representation is generated. The accuracy of the generated mesh is improved by moving its vertices onto the surface implicitly defined by the registered range images. With our approach, it is possible to reconstruct large and complex objects of arbitrary topology out of multiple range images.

Reconstruction has various applications in computer graphics and virtual reality. The geometry of real-world objects can be documented, e.g. statues before and after restoration, or designer models can be reconstructed in the rapid prototyping process. With methods well-known in computer graphics, the appearance of the reconstructed objects can be changed or simulated in different environments.

2. Previous Works & Drawbacks

The registration of range images is based mainly upon two approaches: relational matching and minimization techniques. Relational matching techniques try to find correspondences of previously segmented image primitives such as significant points, edges or planar surfaces [1] [2]. Under the enormous number of possible assignments, the correct matching is searched with tree search methods. The relative orientations of the range images are computed from the corresponding image primitives. In order to make the registration robust and to limit the computation time, the number of image primitives should be small. Thus, the calculated relative orientation is

stressed with a certain inaccuracy.

In order to obtain a better registration accuracy, minimization techniques are basically better for yielding the relative orientation between the range images more accurately. Thereby, a distance metric between two range images is defined and minimized as a function of the unknown location. In general, this is a nonlinear problem. An initial estimate of the unknown location must be computed with other methods. In [3], the distance metric is based on point correspondences. The *iterated-closest-point* algorithm finds the nearest positions on one surface to a collection of points on the other surface and then transforms one surface so as to minimize the distance metric. In [4] and [5], the distance metric is based on correspondences between points on the surface and nearby tangent planes on the other surface. A rigid motion is found that minimizes the point-to-tangent collection directly and then iterates.

The problem of building complete models from a number of range images is considered by Turk and Levoy [6]. The integration and registration of the range images is done step-by-step. In all stages of the algorithm, the range images are represented as triangulated meshes. A modified closest-point algorithm is used to register the range images. Jump edges and self-occlusions are detected when the distance between neighbored vertices of the mesh exceeds a certain threshold. This heuristic criterion requires that the uncertainties of the distance values are smaller than the spacing between the samples. To avoid accumulating the registration errors, a cylindrical range image of the complete object must be captured, and then all other ranges are aligned to this cylindrical view. The main drawback of this approach is that the 3d sensor must be able to capture a cylindrical view of the complete object.

In [7], Bergevin, Laurendeau and Poussart present an approach where the partial range views are represented as triangulated meshes. Features characterizing the triangles are used for a relational matching technique. The meshes are represented in different resolutions to reduce the number of potential correspondences. Multiple trials about correct matching are followed and the best are selected for the generation of better assumptions. The initially found locations are improved by a minimization technique. It is uncertain how many assumptions are necessary to generate the correct result. The results demonstrate that, even for small objects, many initial assumptions must be followed.

In [8], a feature-based approach based on the curvature of the surfaces is also used. The images are mapped on a spherical image, and the transformations between the views are computed by matching the spherical images. In this approach, the objects may not contain holes. For large objects, it is not possible to map a significant part of all range images onto the sphere.

Assuming that all acquired range images are already registered Curless and Levoy present in [9] an volumetric approach to reconstruct complete objects. Each individual range image is converted to a distance function which is stored in a volumetric array. The distance functions for all range images are then combined.

Fig. 1. The processing pipeline.

The object is reconstructed by extracting isosurfaces from the volume grid. A hole filling algorithm is applied in order to interpolate unseen portions of the surfaces.

A signed distance function based on the range images itself is defined by Hilton et al. [10]. The sampling density is arbitrary, but non measured parts of the surface appear as holes in the reconstructed mesh. The mesh itself is generated with a modified marching-cube algorithm. A similar but more stable distance function was developed in [11]. Here, only trusted points which are supported by their local 3d neighborhood are taken into account.

None of the known approaches does a simultaneous registration of all range images. Thus, there is a great need for the registration algorithm presented in section 5.2. In contrast to other approaches, we adapt the range images optimally to each other. Hence the derived distance function in section 8 can be kept simple because artifacts are already eliminated and noise is reduced. By converting the distance function to an intermediate volumetric model and by post-processing the volume, the object can be unambiguously reconstructed.

3. The Processing Pipeline

In our approach, the reconstruction process is divided into several steps (see figure 1). First, the object is scanned from several directions with a 3d sensor. Then,

the acquired range images are segmented with threshold based methods. We discriminate 3 different classes: object, background and undefined regions (shadows, artifacts). It should be noted that nowadays most of the 3d sensors are able to detect shadows automatically, and, that a segmentation into object and background is normally not needed. Therefore, the segmentation only requires little interaction, if any. However, in some limited cases, it is helpful to manually mask regions in the range images which do not contain helpful information for the reconstruction process.

After the segmentation, the relative orientations of all range images are computed in two steps. In the first step, interactively marked corresponding points define a rough estimate of the required orientation. In the second step, the exact relative orientation is determined automatically with a least-squares surface registration with sub-pixel accuracy. By calculating a correction for each measured depth value, all overlapping surfaces are adapted optimally to each other, and the image quality is improved because redundancies are exploited. It is guaranteed that there remain no gaps between the overlapping surfaces. All range images are mapped into a common world coordinate system in order to sculpture a 3d volume which approximates the shape the object. The marching cube algorithm is applied to resample the object geometry in 3d space. The output of the marching cube is improved by automatically moving the vertices of the mesh onto the surface defined by the original range images.

4. 3D Scanning

In the first step, the object is acquired from several directions using a range image finder. For our statue example the range images are captured for example with a structured light range finder which can be easily adjusted for different viewing directions. Each relevant part of the object surface should be visible in at least one view. In order to allow for a registration, neighboring views should show some overlap; typically 20% — 50% overlap is sufficient. Shadows are detected automatically by the 3d sensor. A segmentation of the object from the background can be easily done by threshold-based methods.

5. Registration of Multiple Range Images

5.1. *Point-Based Registration*

The range images are successively integrated into the world coordinate system step-by-step. Starting with the first image, the user selects a new image to be integrated next. The rough relative orientation between the new image and the world coordinate system is defined with at least three corresponding points between the new image and the previously integrated images. The necessary correspondences are selected interactively by the user. The detection of the relative orientation with this approach is described e.g. in [2]. The accuracy of this point identification procedure

need only be very crude: the subsequent steps still work with orientational errors of 30 degrees, or even more.

The rough estimate of the relative orientation is improved by applying the algorithm in section 5.2. Since the relative orientations of all previously integrated images are already known, the algorithm can be slightly modified in order to run much faster in this phase. All interrelationships between the previously integrated images can simply be skipped in the described algorithm, and, the corrections to their relative orientations can be set to be equal. The process of integrating new images to the world coordinate system is repeated with each new range image. After integrating the last range image, the described algorithm in section 5.2 should be performed as described to further improve the relative orientations of all range images.

5.2. *Simultaneous Registration of Multiple Range Images*

The registration of multiple range images $S_0, \dots, S_{n-1}$ is a minimization problem where a distance measurement between the overlapping surface portions is minimized. In our approach, the *registration of all range images is done simultaneously in order not to accumulate registration errors*. For this reason, the relative orientations of all range images must be roughly known initially.

Fig. 2. The relative orientation for each range image with respect to a world coordinate system.

The relative orientation $\vec{\theta}_i$ for each image S_i is defined according to a common world coordinate system. The orientation vector $\vec{\theta}_i$ consists of 6 parameters: 3 rotation angles (the Euler angles) and 3 translation components. The rotation matrix $\mathcal{R}_i^{(\theta)}$ and the translation vector $\vec{t}_i^{(\theta)}$ are derived directly from the parameter vector $\vec{\theta}_i$. The rotation and translation of points $\vec{x} \in S_i$ to the world coordinate system are defined with the operator $T_i^{(\theta)}$.

$$T_i^{(\theta)} \vec{x} \stackrel{\text{def}}{=} \mathcal{R}_i^{(\theta)} \vec{x} + \vec{t}_i^{(\theta)} \quad (1)$$

Now, we define a matching distance $D(S_i, S_j)$ by measuring the distance between

the rotated and translated surfaces S_i and S_j . Let us call the surface S_i the model and the surface S_j the scene. This matching distance has to be minimized as a function of the relative orientations $\vec{\theta}^T = (\vec{\theta}_0^T, \dots, \vec{\theta}_{n-1}^T)$. After minimization, the rotated and translated surfaces should become almost identical in the overlapping areas, i.e. the total residual error ε^2 should become minimal.

$$\varepsilon^2 = \min_{\vec{\theta}} \sum_{i \neq j} D(T_i^{(\theta)} S_i, T_j^{(\theta)} S_j) \quad (2)$$

Since the rotation matrix depends nonlinear on the rotation angles and the range images itself are considered as arbitrary functions over the image grid, the formulated optimization problem is a highly nonlinear problem and cannot be solved analytically. Thus, it must be solved numerically with correction methods.

Fig. 3. The principle of iterative correction.

As shown in figure 3, correction methods start with an initial estimate which is iteratively improved by a correction vector $\vec{\delta}$. In our problem, the unknown mappings $T_i^{(\theta)}$ can be split into a known part $T_i^{(\tau)}$ and a correction $T_i^{(\delta)}$ according to $T_i^{(\theta)} = T_i^{(\delta)} T_i^{(\tau)}$.

$$\varepsilon^2 = \min_{\vec{\delta}} \sum_{i \neq j} \varepsilon_{ij}^2 \quad (3)$$

$$\varepsilon_{ij}^2 = D(T_i^{(\delta)} T_i^{(\tau)} S_i, T_j^{(\delta)} T_j^{(\tau)} S_j) \quad (4)$$

The correction vector $\vec{\delta}$ is searched in the local neighborhood of $\vec{\tau}$ by linearizing the problem at the estimated solution $\vec{\tau}$. For this reason, the model S_i appearing in the matching distance is represented piecewise by its tangential planes $\vec{n}_{ik}(\vec{x} - \vec{x}_{ik}) = 0$, where k is the index of the available model points. The normal vectors $\vec{n}_{ik}$ are simply computed with local gradient operators in the range image S_i at the measured point $\vec{x}_{ik}$. $T_i^{(\tau)}$ can be effectively applied by rotating and translating the tangential plane:

$$\vec{n}_{ik}^{(\tau)} = \mathcal{R}_i^{(\tau)} \vec{n}_{ik} \quad \vec{x}_{ik}^{(\tau)} = T_i^{(\tau)} \vec{x}_{ik} \quad (5)$$

A corresponding scene point $\vec{y}_{ijk}^{(\tau)}$ can be easily found by mapping the point $\vec{x}_{ik}^{(\tau)}$ into the coordinate system of the range image S_j and then projecting it onto the parameter grid of the image S_j (see Fig. 4). In the parameter grid of the

Fig. 4. Projecting the model point to the scene yields the corresponding scene point.

scene image S_j , the depth is interpolated and the interpolated point $\vec{y}_{ijk}$ has to be mapped back to the world coordinate system.

$$\vec{y}_{ijk}^{(\tau)} = T_j^{(\tau)} \vec{y}_{ijk} \quad (6)$$

For most of the points, the distance between the corresponding points $\vec{x}_{ik}^{(\tau)}$ and $\vec{y}_{ijk}^{(\tau)}$ should be small. Exceptions to this rule can be detected with the outlier test in section 6.4. The outliers have to be rejected from the registration algorithm. With these definitions, the distance between the model and the scene is expressed as

$$\varepsilon_{ij}^2 = \sum_k \varepsilon_{ijk}^2 \quad (7)$$

$$\varepsilon_{ijk} = \mathcal{R}_i^{(\delta)} \vec{n}_{ik}^{(\tau)} \left(\left(\mathcal{R}_j^{(\delta)} \vec{y}_{ijk}^{(\tau)} + \vec{t}_j^{(\delta)} \right) - \left(\mathcal{R}_i^{(\delta)} \vec{x}_{ik}^{(\tau)} + \vec{t}_i^{(\delta)} \right) \right) \quad (8)$$

After linearization and some algebraic manipulations, we obtain for each point k a $1 \times 6n$ matrix $\mathcal{A}_{ijk}$ and a scalar s_{ijk} . With these variables the total residual ε^2 can be expressed as

$$\varepsilon^2 = \min_{\vec{\delta}} \sum_{i \neq j, k} \varepsilon_{ijk}^2 = \min_{\vec{\delta}} \sum_{i \neq j, k} \|\mathcal{A}_{ijk} \vec{\delta} - s_{ijk}\|^2 \quad (9)$$

with

$$s_{ijk} = \vec{n}_{ik}^{(\tau)} \cdot (\vec{x}_{ik}^{(\tau)} - \vec{y}_{ijk}^{(\tau)}) \quad (10)$$

$$A_{ijk} = \begin{pmatrix} \underbrace{0 \dots 0}_{6i \times 1} & \underbrace{C_{ijk}}_{6 \times 1} & \underbrace{0 \dots 0}_{6(l-i-1) \times 1} \end{pmatrix} + \begin{pmatrix} \underbrace{0 \dots 0}_{6j \times 1} & \underbrace{-C_{ijk}}_{6 \times 1} & \underbrace{0 \dots 0}_{6(l-j-1) \times 1} \end{pmatrix} \quad (11)$$

$$C_{ijk} = \begin{pmatrix} \vec{n}_{ik}^{(\tau)} \times \vec{y}_{ijk}^{(\tau)} \\ -\vec{n}_{ik}^{(\tau)} \end{pmatrix}^T \quad (12)$$

6. Solving the Nonlinear Least-Squares Problem

Since the world coordinate system can be freely defined, there is a linear dependency between the unknown parameters. This dependency can be avoided by defining that the relative orientation of the first range image is identical to the orientation of the world coordinate system. Based on this assumption, the derived linearized equation system can be easily reduced. The remaining problem can be solved with standard methods. With the Newton-Taylor algorithm, the correction vector $\vec{\delta}$ is computed with

$$\vec{\delta} = \left(\sum_{i \neq j, k} \mathcal{A}_{ijk}^T \mathcal{A}_{ijk} \right)^{-1} \sum_{i \neq j, k} (\mathcal{A}_{ijk}^T s_{ijk}) \quad (13)$$

In fact, we use the Levenberg-Marquardt method to solve the numerical problem [12] [13]. It differs from the Newton-Taylor method in two points. First, the expected magnitude of all unknowns is made equal by substituting the unknowns with scaled ones. The square of good scaling factors are found on the main diagonal of the normal matrix $\mathcal{N} = \sum_{i \neq j, k} \mathcal{A}_{ijk}^T \mathcal{A}_{ijk}$. Second, a multiple of the unit matrix is added to the normal matrix. With a suitable chosen multiple of the unit matrix, it can be guaranteed that ε^2 decreases in each iteration and that numerical problems never occur.

6.1. Statistical Considerations

The statistical regression model allows an estimation of the mean deviation σ_i of the components $\vec{\delta}$. A reasonable assumption is that the residuals ε_{ijk} are uncorrelated normal random variables with mean 0 and variance σ^2 . A reliable estimate of σ^2 is given by the normalized total residual error. In order to get a reliable estimate, the denominator of the quotient has to be decreased by the number of degrees of freedom. If m specifies the number of all equation rows, σ^2 is given by:

$$\sigma^2 = \varepsilon^2 / (m - 6n) \quad (14)$$

The corresponding covariance $\Sigma_{\delta\delta}$ can be computed effectively in (15):

$$\Sigma_{\delta\delta} = \sigma^2 \left(\sum_{i \neq j,k} \mathcal{A}_{ijk}^T \mathcal{A}_{ijk} \right)^{-1} \quad (15)$$

Then, the corresponding variances σ_i^2 are found on the main diagonal of the covariance matrix.

6.2. Termination Criterion

In each iterative algorithm, the termination criterion plays an important role. The algorithm may terminate if all components of the correction vector $\vec{\delta}$ are lower than a predefined threshold. We developed a better method in our approach. Our test is based on a statistical test for the hypothesis $\|\vec{\delta}\| = 0$. The algorithm terminates if the correction has no statistical relevance and is only caused by noise.

Therefore, we need a significance test which verifies the hypothesis $\|\vec{\delta}\| = 0$. Under the assumption that the residuals ε_{ijk} in Eq. (9) are normal random variables and that the hypothesis $\|\vec{\delta}\| = 0$ is true, the variable t in (16) has a χ_{6n}^2 -distribution. For the test, the components of the correction vector $\vec{\delta}$ are normalized by their expected inaccuracies:

$$t = \sum_i \frac{\delta_i^2}{\sigma_i^2} \quad (16)$$

With $t < 6n$, we have a statistical test for the above hypothesis. If the test is true, no more iterations of the nonlinear least-squares problem are needed. In practice, we need at least $8n + 40$ points to gain statistically reliable results for the test variable t .

6.3. Resolution Hierarchy

The robustness and the efficiency of the proposed algorithm can be improved by introducing a resolution hierarchy. In the first stages of the interactive process, only a few points of the model (according to the notation in section 5.2) are used. In subsequent stages, additional model points are included in the computation process. In practice, the chosen points in the lower resolutions should be distributed uniformly in the image plane. Thus the resolution hierarchy can be simply applied just by choosing only those points in the image grid whose grid coordinates are a power of 2^d . However, the corresponding scene points should be found in the full resolution of the range images. Each level in the resolution hierarchy is maintained until the statistical termination criterion is fulfilled. Then, the resolution is doubled in each dimension. In this way, the algorithm quickly gains the full resolution of the 3d model. In the finest resolution, the computed location parameters reach their maximal accuracy. The speed-up of this simple resolution hierarchy is a factor of

20 – 100 because the run time of the registration depends mainly on how often the registration algorithm is iterated in the highest resolution.

6.4. Elimination of Outliers

Missing parts, occlusions and scan errors influence the accuracy of the registration algorithm. Especially in the first iterations, the initial estimate of the relative orientation is very inaccurate. Thus, the rotated and translated surfaces do not fit very well together. By eliminating outliers, most of the occluded or missing parts are automatically detected. With the same procedure, another important problem which arises during the registration can be solved. All pairs of model points $\vec{x}_{ik}^{(\tau)}$ and their corresponding scene points $\vec{y}_{ijk}^{(\tau)}$ which do not belong to each other must be rejected from the registration.

Outliers of the model points cause a residual ε_{ijk} which lies above the average of the other residuals. Normally, the average of the residuals are computed at the end of each iteration. Fortunately, there is a very efficient way to detect outliers earlier. In each iteration of the applied correction method, the total residual and the mean residual σ of the last iteration is known. s_{ijk} in (10) specifies the residual belonging to the previous iteration. Thus, outliers are detected with the test

$$|s_{ijk}| > 3\sigma_{\text{last}} \quad (17)$$

However, in practice, it is preferable to perform additional outlier tests, e.g. a test depending on the directions of the normal vectors or, simply, the following test:

$$\left| \vec{x}_{ik}^{(\tau)} - \vec{y}_{ijk}^{(\tau)} \right| > 3\sigma_{\text{last}} \quad (18)$$

These tests are very efficient, because the values of the variables must be computed in the registration algorithm anyway. If some model points are eliminated in one iteration, the same points can be taken into account in the next iterations. Thus, this test is robust with respect to faulty decision-making in previous iterations. Statistical considerations show that the outlier test does not influence the convergence of the registration algorithm. This can be shown by considering the iterated outlier test. The number of eliminated outliers converges very fast. In the case of normal distributed residuals, approximately 1% of the model points are eliminated in the infinitely often iterated outlier test.

6.5. Improvements

The registration of multiple range images is a highly nonlinear problem with several small local minima besides the global minimum. Hence, the iterative algorithm of the nonlinear least-squares problem can converge in a small local minimum. The risk of this behavior can be reduced by smoothing the total residual as a function of the unknown parameters. As a result, small local minima disappear. This smoothing can be performed by averaging the normal vectors of the model in their local neighborhood. Of course, such a smoothing influences the accuracy of

the found location parameters. Hence, the smoothing should be applied in the first iterations. In subsequent stages, the smoothing should be reduced successively.

7. Adapting Overlapping Surfaces

Due to measurement noise and small inaccuracies of the 3d sensor calibration the registered range images do not fit perfectly to each other. In order to simplify the reconstruction and to enhance the results it is necessary to adapt the overlapping surfaces, while outliers are eliminated. Thus we search for every image pixel $f_i(u, v)$ a correction value.

Fig. 5. (a) Finding intersections. (b) The scalar product between the surface normal and the scanning direction specifies the density of the point measurements.

For each measured pixel $\vec{x}$ of a range image we define a ray through $\vec{x}$ along the viewing direction $\vec{c}$ of the range finder device. Note that all points on this ray produce the same grid coordinates (u, v) when projecting them back onto the image grid. Now the intersection between this ray and all other overlapping range images is calculated. The ray is mapped into the coordinate system of the respective range image $f_j(u, v)$. If the start point derived from $\vec{x}$ is below the surface, an upper intersection point is searched in the direction $\vec{c}$ to the camera. If the start point is above the surface, a lower intersection point is searched for. First, the intersection may be bracketed by finding an upper and a lower bound of the intersection point. Then the intersection can be found exactly by applying a bisection method or a combined bisection/secant method. In practice, the total search space may be bounded by a multiple of the measurement noise σ , calculated in Eq. (14) during the registration. The search can be stopped as soon as the intersection point is found or the projection of the point onto the image grid yields an undefined value.

As a result of the ray casting algorithm, a displacement λ of the point $\vec{x}$ along the ray is obtained. $\lambda\vec{c} + \vec{x}$ notifies the intersection point in the coordinate system of the processed range image. The potentially best correction can be obtained by a robust combination of all displacements $\langle \lambda_k \rangle$ calculated for all overlaps. Therefore, we introduce a number of weights for each intersection point.

The scalar product between the surface normal $\vec{n}$ and the scanning direction $\vec{c}$ specifies the density of the point measurements. In order to eliminate jump edges the following weight w_g is defined:

$$w_g(u, v) = \begin{cases} \vec{n}(u, v) \cdot \vec{c}(u, v) & \text{if } \vec{n}(u, v) \cdot \vec{c}(u, v) \geq 0.1 \\ 0 & \text{if } \vec{n}(u, v) \cdot \vec{c}(u, v) < 0.1 \end{cases} \quad (19)$$

Since the combined range images should build a C^1 -smooth surface in the overlapping parts, a second weight w_b is used. It approximates the shortest quadratic distance from the actual grid point to the outline of the surface portions, successfully measured in the image where the intersection is found. It can be efficiently calculated with the well known Chamfering algorithm [14].

$$w_b(u, v) = \min_{(\hat{u}, \hat{v})} \{ (u - \hat{u})^2 + (v - \hat{v})^2 \mid f(\hat{u}, \hat{v}) \text{ is not defined} \} \quad (20)$$

Now we can combine all displacements $< \lambda_k >$ by calculating their weighted average μ_λ and standard deviation σ_λ . In order to eliminate outliers the calculation of the average is iterated. Each data is assigned an additional weight according to Tukey's biweight function in Eq. 21. The total weight is given in Eq. (22).

$$w_t(\xi) = \begin{cases} (1 - (\xi/c_t)^2)^2 & \text{if } |\xi| \leq c_t \\ 0 & \text{if } |\xi| > c_t \end{cases} \quad \text{with } c_t = 4.6851 \quad (21)$$

$$w(u, v, \lambda_k) = w_t((\lambda_k - \mu)/\sigma) w_g(u, v) w_b(u, v) \quad (22)$$

Tukey's biweight function even suppresses the outliers. The 95% asymptotic efficiency on the standard normal distribution of the Tukey's biweight function is obtained with the tuning constant $c_t = 4.6851$. The algorithm converges very fast after two or three iterations. It eliminates outliers and guarantees a smooth overlap between the different range images. After applying this procedure to all measured surface points, the images are optimally adapted to each other.

8. Visibility Criterion

In contrast to other approaches where the range images are typically considered as a set of points, we consider the range images as partially defined surfaces. By interpolating between the grid points the surface can easily be continuously completed. Our intention is to convert the range images to a signed distance function. Previously distance functions have been developed [10] [11] for the implicit definition of a surface. We here present an alternative of which the rational is the fact that arbitrary points lying below the surface in a range images are invisible from the 3d sensor, whereas points lying between the surface and the 3d sensor are visible, and, thus, must not belong to the object. Thus, all interpolated points in a range image lying between the surface and 3d sensor of another range image cannot be used for the definition of the surface (see fig. 6).

Fig. 6. Here P_2 is interpolated in view 2. But in the sense of the visibility criterion P_2 is visible in view 1 and hence cannot belong to the object. This way the visibility criterion resolves self-occlusions.

This leads us to the idea that the object surface can be implicitly defined as a function $h(\vec{x}) = 0$ where $\vec{x}$ is a point in 3d space. For the definition of $h(\vec{x})$ we define for each range image f_i a function $g_i(x)$ which measures the signed distance between a point in space and the interpolated surface in $f_i(u, v)$. The corresponding point in the range image $f_i(u, v)$ is found by projecting the point $\vec{x}$ onto the parameter grid of the range image. Thus, a positive distance indicates that the point lies between the 3d sensor and the surface whereas a negative distance indicates that the point lies below the surface. If the range image $f_i(u, v)$ is undefined at the requested position, the function $g_i(\cdot)$ will be undefined as well. The synthesis of multiple range images is achieved by combining the functions $g_i(\cdot)$ as follows:

$$h(\vec{x}) = \max_i \{ g_i(\vec{x}) \} \quad (23)$$

Note, that $h(\vec{x})$ only presents a partially defined function in 3d space. There remain undefined points due to the 3d sensor only capturing subsections of the measured volume. We can easily complete the function $h(\cdot)$ by defining $h(\vec{x}) = 1$ for all points $\vec{x}$ where none of the functions g_i are defined. With the implicit defined surface $h(\vec{x}) = 0$ only the measured parts of the surface are defined, but the zero crossings of $h(\vec{x})$ define the surface even in unseen parts of the surface. The point $\vec{x}$ is called visible if

$$h(\vec{x}) \geq 0 \quad (24)$$

Now, this visibility criterion is used to reconstruct the object unambiguously even in the case of complicated topologies. After registration and estimation of the relative orientation, the partial views of the range images are used to carve out an intermediate volumetric model of the object. This is done by applying the visibility criterion in Eq. (24) for each voxel of the volume (see figure 7).

Fig. 7. Carving out the volume.

In some cases, for some voxels lying far away from the object, it is not possible to directly decide whether they belong to the object. Figure 8 illustrates this. If the overlapping scan views (view 1 and 2) are captured under a certain angle, points far away from the object may lie outside the scan views. Assume that there is another scan view (scan view 3) capturing the back face of the object. Then, the point P is not visible in the sense of criterion (24). The criterion (24) alone cannot decide if P belongs to the object.

Fig. 8. Undefined points in 3d space.

This problem is solved with a post-processing step on the volume. At first, all connected volumetric segments have to be detected in the volume. Secondly, a significant number of randomly chosen surface points, say 1000 – 10000, are mapped back into the volume grid. For each volume segment the number of points falling in their local neighborhood is counted. All but the segment which received most of the hits are deleted. In addition to the deletion step, closed, hollow spaces in the remaining volume are completely filled. The intermediate volume describes the topology of the object and approximates its shape within the desired level of detail. It accelerates the reconstruction process and allows the reconstruction of objects

without any topological constraints.

With this intermediate volume, the visibility criterion can be completed. A point P in 3d space is called visible if the criterion (24) is true. If the criterion (24) is false, then the intermediate volume must decide the visibility. If the point P falls into a cube of the volume where the majority of the bordering voxels vote for visibility, then the point P is said to be visible; otherwise, it is marked as invisible. Normally, all of the bordering voxels have an equal status. If not, this tells us that some parts of the surface are not captured with the range finder, or the overlap of the scanning views is smaller than the size of one cube in the intermediate volume.

9. Generating the Polygonal Model

After the generation of the volumetric model, the marching cube algorithm with a look-up table that resolves ambiguous cases (see figure 9) can be applied to generate a polygonal representation [15] [16]. The accuracy of this polygonal mesh is improved by moving the vertices of the mesh onto the surface implicitly defined by the registered range images. The point is moved between the voxels of the cube until the zero crossing of the signed distance function in Eq. (24) is found. The exact coordinates of the moved vertices are interpolated in the range images. This step is similar to a ray casting algorithm which is combined with the visibility criterion. This way, the full accuracy of the scanning device is exploited at all vertices of the mesh, and concave object shapes are also modeled correctly. The amount of polygons can be significantly reduced by applying a polygon reduction [17] [18].

Fig. 9. Triangulation of a cube by applying Marching-Cubes and improving the mesh.

10. Results

Results are shown for a molar tooth. The range images of the tooth are captured with a laser scanner based on the principle of triangulation. The scanning of one view (256.000 points) takes about 45 seconds. The distance between two scanned points is $25\ \mu\text{m}$ and the height resolution is $10\ \mu\text{m}$. In order to obtain the necessary

Fig. 10. Steps through the reconstruction process: polygonal model, rendered model.

accuracy, the triangulation angle is quite large. Consequently, there are obviously a lot of shadow areas. In figure 10 a low resolution polygonal model and the high resolution rendered image are shown. The tooth was reconstructed out of 25 range images acquired from different scanning views. The interactive reconstruction steps are performed within a few minutes, say 20 seconds per range image. The simultaneously registration takes 11.5 minutes on a Ultra-SPARC, and the adaption of the overlapping surfaces takes 16 minutes. The run times for sculpturing and post-processing the volume are shown in table 1. In total 8 minutes were spent for the interactive work and 36 minutes for the automatisms.

Table 1. Run times for sculpturing, post-processing the volume and polygonalization. The times are given in minutes:seconds and are measured on a Ultra-SPARC.

Object	Scans	Volume Size	Sculpt./Postp.	Polyg.	Triangles
Tooth	25	$170 \times 200 \times 193$	7:07	1:03	392764
Statue	285	$500 \times 217 \times 221$	92:36	9:49	718904
Statue	285	$700 \times 304 \times 309$	295:31	22:40	1434296

Another example is presented of a statue 182cm tall. The statue was scanned with a structured light sensor which can be easily adjusted for different viewing angles. Each range image shows a window of approximately $40 \times 40 \text{ cm}^2$. The statue has a complicated topology, because it contains holes and many occlusions. Due to its size and complicated geometry, the statue was scanned from 285 views from all directions (see figure 11). All range views are combined with the described method. This takes approximately one day. The overall registration error is 0.581mm before exploiting redundancies with the method described in section 7. Afterwards, the

remaining root mean square of all residuals is reduced to 0.068mm . This remaining error can be fully explained by interpolation errors in the image grid. The adaption of the overlapping surfaces takes 706 minutes. The original and the reconstructed statue are shown in figure 12. Some range images and the reconstruction program are shown in figure 11.

Fig. 11. Modeling a statue from 285 range images. The range images are displayed with a simple rendering technique.

11. Conclusions and Future Work

One main contribution of our approach lies in the simultaneous and robust registration of all range images with sub-pixel accuracy. This way, it is even possible to reconstruct large objects from numerous small range images. The registration process is based on a least-squares approach where a distance metric between the overlapping views is minimized. A resolution hierarchy accelerates the registration process. The registration process is robust even in the presence of small artifacts that typically appear in range images.

While adapting the overlapping surfaces the image quality of the range images is significantly improved. Redundancies are exploited and the reconstruction accuracy

Fig. 12. Original (left) and reconstructed statue. The statue is 182cm tall and was reconstructed out of 285 range images, each showing approximately a $40\text{cm} \times 40\text{cm}$ portion of the statue.

in the overlapping parts is typically better than the accuracy of the original range images. It is guaranteed that there are no gaps between the overlapping parts of the range images. With the developed signed distance function and visibility criterion, it is possible to decide whether or not each point in 3d space belongs to the object. This knowledge is exploited for the generation of a model. The object can be resampled by generating an intermediate volumetric model. The generated volumetric model can be used for the generation of a polygonal mesh. The resolution of the volume can be chosen in accordance to the desired accuracy of the tessellation.

Further work has to be done to generate an adaptive volume resolution and to generate a polygonal mesh which approximates the original surface in nearly the same quality with less polygons.

References

1. G. Vosselman, *Relational Matching* (Springer, 1992).
2. O. D. Faugeras and M. Hebert, The Representation, Recognition, and Locating of 3-D

- Objects, *The International Journal of Robotics Research* **5** (1986) 27–53.
3. P. J. Besl and N. D. McKay, A Method of Registration of 3-D Shapes, *IEEE Trans. Pattern Analysis and Machine Intelligence* **14** (1992) 239–256.
 4. P. J. Neugebauer, “Hochgenaue Objektlokalisierung in Tiefenbildern”, in *Visualisierung — Rolle von Interaktivität und Echtzeit*, Sankt Augustin, Germany (June 1992).
 5. Yang Chen and Gérard Medioni, Object modelling by registration of multiple range images, *Image and Vision Computing* **10** (1992) 145–155.
 6. G. Turk and M. Levoy, “Zippered polygon meshes from range images”, in *Proc. of SIGGRAPH '94*, Orlando, Florida (1994) pp. 311–318.
 7. R. Bergevin, D. Laurendeau, and D. Poussart, Registering range views of multipart objects, *Computer Vision and Image Understanding* **61** (1995) 1–16.
 8. K. Higuchi, M. Hebert, and K. Ikeuchi, Building 3-d models from unregistered range images, *Graphical Models and Image Processing* **57** (1995) 315–333.
 9. B. Curless and M. Levoy, “A volumetric method for building complex models from range images”, in *Proc. of SIGGRAPH '96*, New Orleans, Louisiana (1996) pp. 303–312.
 10. A. Hilton, A. J. Stoddart, J. Illingworth, and T. Windeatt, “Reliable surface reconstruction from multiple range images”, in *4th European Conf. on Computer Vision*, (Springer, 1996), pp. 117–126.
 11. M. D. Wheeler, Y. Sato, and K. Ikeuchi, “Consensus Surfaces for Modeling 3D Objects from Multiple Range Images”, in *DARPA Image Understanding Workshop*, New Orleans, Louisiana (1997).
 12. D. W. Marquardt, An Algorithm for Least-Squares Estimation of Nonlinear Parameters, *J. Soc. Indust. Appl. Math.* **11** (1963) 431–441.
 13. C. L. Lawson, R. J. Hanson, *Solving Least Squares Problems* (Prentice-Hall, Englewood Cliffs N.J., 1974).
 14. G. Borgefors, “Chamfering: A Fast Method for Obtaining Approximations of the Euclidean Distance in N Dimensions”, in *3rd Scand. Conf. on Image Analysis*, Copenhagen, Denmark (1983).
 15. W.E. Lorensen and H. E. Cline, “Marching cubes: A high resolution 3d surface construction algorithm”, in *Proc. of SIGGRAPH '87*, Anaheim, California (1987) pp. 163–169.
 16. C. Montani, R. Scateni, and R. Scopigno, A modified look-up table for implicit disambiguation of marching cubes, *Visual Computer* **10** (1994).
 17. W. Schroeder, J. Zarge, and W. Lorensen, “Decimation of triangle meshes”, in *Proc. of SIGGRAPH '92*, Chicago, Illinois (July 1992) pp. 65–70.
 18. H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle, “Mesh optimization”, in *Proc. of SIGGRAPH '93*, Anaheim, California (1993) pp. 19–26.