LaserBrush: A Flexible Device for 3D Reconstruction of Indoor Scenes

Martin Habbecke and Leif Kobbelt
Computer Graphics Group, RWTH Aachen University
http://www.graphics.rwth-aachen.de

Abstract

While many techniques for the 3D reconstruction of small to medium sized objects have been proposed in recent years, the reconstruction of entire scenes is still a challenging task. This is especially true for indoor environments where existing active reconstruction techniques are usually quite expensive and passive, image-based techniques tend to fail due to high scene complexities, difficult lighting situations, or shiny surface materials. To fill this gap we present a novel low-cost method for the reconstruction of depth maps using a video camera and an array of laser pointers mounted on a hand-held rig. Similar to existing laser-based active reconstruction techniques, our method is based on a fixed camera, moving laser rays and depth computation by triangulation. However, unlike traditional methods, the position and orientation of the laser rig does not need to be calibrated a-priori and no precise control is necessary during image capture. The user rather moves the laser rig freely through the scene in a brush-like manner, letting the laser points sweep over the scene's surface. We do not impose any constraints on the distribution of the laser rays, the motion of the laser rig, or the scene geometry except that in each frame at least six laser points have to be visible. Our main contributions are twofold. The first is the depth map reconstruction technique based on irregularly oriented laser rays that, by exploiting robust sampling techniques, is able to cope with missing and even wrongly detected laser points. The second is a smoothing operator for the reconstructed geometry specifically tailored to our setting that removes most of the inevitable noise introduced by calibration and detection errors without damaging important surface features like sharp edges.

1 Introduction

Three dimensional geometry has become a common media type in recent years mostly due to the availability of powerful and affordable graphics hardware and the decreasing costs for storage and transmission of large amounts of data. However, the generation of 3D models is still the main obstacle preventing an even more widespread use of geometry-enhanced applications: Manual design of 3D models is time consuming and tedious, automatic methods are either extremely costly if, e.g., commercially available laser scanners or structured light systems are used, or are extremely difficult, error-prone and limited to specific scenarios if more recent image- and video-based methods are used. This is especially true for the case of rooms and indoor scenes. Most commercially available digitizing solutions are specialized on close-range operation, i.e., on the recovery of objects rather than whole rooms. Image-based binocular or multi-view stereo methods easily fail in indoor scenarios due to the lack of sufficient information for photo consistency computation.

We present a novel depth map estimation method targeted at the reconstruction of indoor scenes that is easy to use and implement. The hardware requirements are limited to an off-the-shelf digital video camera and a set of low-cost laser pointers. The lasers are rigidly mounted on a rig such that they cast an irregular set of rays into a scene or onto an object (cf. Figure 1 for a picture of our prototype laser rig with 20 lasers). Similar to traditional laser scanning techniques, the video camera is positioned on a tripod and observes the laser points in the scene. Due to the irregular configuration of the laser rays, the position and orientation of the laser rig does not have to be calibrated a-priori or precisely controlled during the depth recovery process. We rather let the user move the laser rig freely through the scene in a brush-like manner and recover its position and orientation from the observed laser points for each frame. Depth values are then computed by triangulation. The main problem of active laser- or light-based reconstruction systems is that of occlusion: concave parts of the surface can only be reconstructed as long as *both*, the camera and the light source have an unoccluded view on the surface. The flexibility of our approach relaxes this problem significantly since the user is free to cast laser rays from any direction into the scene. Hence concave parts of the surface can be recovered as long as the camera has an unoccluded view.

The limited number of lasers, errors in the calibration of the laser rays, as well as imperfect detection of laser points in the input images may induce a notable level of noise in the resulting depth maps. We show empirically that a large portion of this noise has a systematic character and develop a smoothing operator specifically tailored to our setting that is able to remove most of the noise without sacrificing important surface features.

Our algorithm has been developed with the following goals in mind which, in this combination, cannot be found in any of the existing scanning and reconstruction methods:

Robustness. By exploiting robust sampling techniques our algorithm is able to cope with missing and even wrongly detected laser points. Occluded and later reappearing laser points are picked up again and assigned to the correct laser ray.

Versatility. Our method does not rely on any specific structure of the scene like known angles or additional cues. Hence it is applicable to a wide spectrum of scenarios ranging from whole scenes to single objects.

Ease of implementation. All image processing, i.e., the detection of laser points and the calibration pattern, is kept as simple as possible without any thresholds that require tedious per-scene adjustment. All energy minimizations rely on standard optimization techniques.

Affordability. All required hardware components are available off-the-shelf at reasonable prices (about 100€ for the laser rig).

2 Related Work

The surveys presented by [Besl 1989] and, more recently, by [Blais 2004] provide a thorough overview of existing active 3D reconstruction methods. For the field of passive, purely image-based

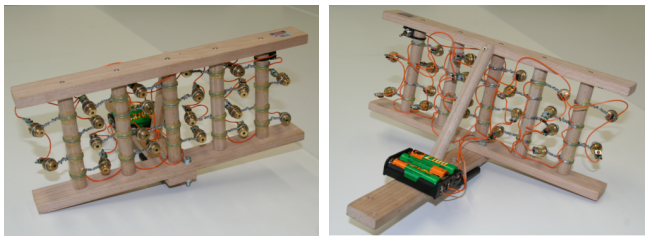

Figure 1: *The hand-held, battery driven laser rig with 20 laser pointers we used for our experiments.*

methods, [Scharstein and Szeliski 2002] have presented an extensive discussion and evaluation of recent binocular stereo methods. A similar effort has been published by [Seitz et al. 2006] for recent multi-view stereo reconstruction methods. In the following we review and compare the subset of reconstruction techniques most relevant to our work.

A popular class of active reconstruction systems uses structured light. That is, a light source or a projector casts a regular pattern onto the surface of the scene, one or more cameras observe the pattern and derive depth information. Examples for structured light systems based on a single camera have been presented by [Rocchini et al. 2001], who use a standard video projector as light source and focus on low overall system cost, and [Pipitone and Hartley 2006] who use a specialized setup with a xenon tube on a turntable. The main difference to our method is the requirement of a precise calibration of the projector's pose with respect to the camera. This calibration is done off-line and has to be repeated every time either the camera or the projector is moved. In our method, we derive the pose of the laser rig for each input image, allowing for a much more flexible freehand operation. Hence the constellation of the laser rays in the rig has to be determined only once. Methods based on stereo cameras exchange the calibration of the projector with the calibration of the stereo setup and then derive depth information by determining corresponding pixels in the stereo images with the help of the projected pattern. [Scharstein and Szeliski 2003] have presented a stereo-based structured light system for high precision, and more recently [Weise et al. 2007] published a system incorporating motion compensation, targeted at the reconstruction of moving persons and objects. See [Rocchini et al. 2001] and [Weise et al. 2007] and the references therein for more details on monocular and binocular structured light systems, respectively. Usually structured light systems are targeted at object reconstruction due to the limited range of the projector and often require dimmed lighting conditions to ensure that the pattern is detected correctly. In contrast, the standard laser pointers our method is based on can be faithfully detected at long distances, on difficult surfaces, and under regular everyday lighting conditions, which enables the snap-shot reconstruction even of large indoor scenes.

Also related, but requiring an even more constrained environment is the approach of [Bouguet and Perona 1998]. They compute depth information from the shadow of a straight object using a calibrated camera and known ground plane. Examples of methods that work with hand-held laser plane emitters are [Winkelbach et al. 2006] and [Zagorchev and Goshtasby 2006]. To determine the pose of the emitter, both methods require additional cues: Winkelbach et al. require two planes enclosing an exactly known angle which are both intersected by the laser plane in every frame, the method of Zagorchev and Goshtasby is based on a double-frame arranged around the object to be reconstructed. Hence, both methods are limited to the reconstruction of objects rather then whole scenes, since such strict requirements on the background or a calibration target cannot easily be met in practice. The methods presented by [Takatsuka et al. 1999] and [Furukawa and Kawasaki 2003] utilize markers in the form of LEDs attached to a laser emitter. With the help of the markers the pose of the laser emitter is determined, allowing for an easy triangulation of depth values, but under the constraint that all markers have to be visible in the images. More recently, [Kawasaki et al. 2006] have presented a calibration-free reconstruction method using two perpendicular laser planes. They recover the emitter poses from laser plane intersections detected in a sequence of input images. The main advantage is that, once the parameters of a plane have been recovered, a whole projected line can be triangulated instead of single image points. The approach does, however, suffer from numerical instabilities especially in the case of non-detectable plane intersections due to occlusions. In comparison, our method generates less samples per frame, but robustly copes with occluded laser points.

Probably most related to our work is the Model Camera of [Popescu et al. 2006]. It uses a laser rig rigidly mounted to a hand-held video camera. By enabling the user to freely move the camera through a scene, this method imposes fewer constraints than all the above systems that rely on a stationary camera. This comes, however, at the cost of reduced stability since estimating relative changes of the camera pose between successive frames is an error prone problem which tends to accumulate errors over a sequence of frames even with the additional information provided by sparse per-frame depth samples. In the follow-up project by [Bahmutov et al. 2006], the authors address this problem and employ a more constrained setup by mounting the Model Camera on a tripod. In addition, two shaft encoders were used to precisely measure pan and tilt angles, thereby stabilizing the pose estimation process but requiring an exact synchronization of the captured images and the angle measurement process. The main drawback, however, is the reduced number of only two degrees of freedom for the camera motion in comparison to the six degrees of the original Model Camera. Since the laser rig is still mounted to the camera, it is difficult to vary the sampling density for different parts of a scene and this setup is subject to the same occlusion problems as standard laser scanners. Hence we decided to constrain the scene capture process as little as possible and to allow the user to operate the laser-brush in a freehand manner in order to be able to deal with arbitrary scenarios.

3 Method Overview

In this paper, we denote laser rays by $L_i(\lambda) := c_i + \lambda r_i$, with $c_i, r_i \in \mathbb{R}^3$, and corresponding laser point positions in image space by $p_i \in \mathbb{R}^2$. The index i always counts per-frame entities, while j is defined to be the frame index. Hence, $p_{i,j}$ denotes the ith image space point in frame j. Points in 3-space are denoted by $q \in \mathbb{R}^3$, and planes by $N \in \mathbb{R}^4$, respectively. The intrinsic parameters of the camera are given in the form of a matrix $K \in \mathbb{R}^{3\times3}$, the back-projected viewing ray of an image point p_i is denoted by $v_i := K^{-1}p_i$. Since the camera is defined to reside in the origin of the world coordinate frame we do not need to take extrinsic parameters into account. The vertices of the resulting surface mesh are denoted by $x_{i,j} \in \mathbb{R}^3$. Finally, images are given as bi-variate functions $I(u, v)$, with (u, v) being pixel coordinates.

Our method consists of the following steps. First the camera's intrinsic parameters K and the laser rig are calibrated as detailed in Section 5. We then mount the camera on a tripod, sweep the laser rays through the scene and capture the resulting laser points $p_{i,j}$ for each frame j. Given the positions of the laser points in a respective frame, the central idea of our method is to find the pose (i.e., position and orientation) of the laser rig in space that minimizes the sum of squared Euclidean distances between corresponding image points p_i and laser rays L_i, projected to image space lines. Once

the pose of the laser rig is recovered, a depth value can be computed for each detected laser point in image space by triangulation. Corresponding laser ray to image point pairs are initialized using a simple heuristic and maintained by tracking laser points over successive frames. We accumulate the depth information over several hundred frames and compute the Delaunay triangulation of all detected points in image space to obtain a quasi-dense depth map. To reconstruct a surface mesh, we transfer the Delaunay triangulation to 3-space using the reconstructed 3D points $x_{i,j}$. Finally we apply a smoothing operator to the surface mesh to remove systematic noise introduced during the estimation of the laser rig's pose (cf. Section 7).

4 Laser Detection and Tracking

The detection of laser points in the input images consists of two main steps. The first is the computation of a difference image, the second the computation of a cross-correlation function. For the difference computation we take a picture I_{empty} of the scene without laser points. Then, for each image I containing laser points, we compute the intensity channel

$$f(u,v) := \big(r_I(u,v) + g_I(u,v)\big) - \big(r_{\text{empty}}(u,v) + g_{\text{empty}}(u,v)\big), \quad (1)$$

where r_I, g_I denote the red and green color channels of I and r_{empty}, g_{empty} denote the respective channels of I_{empty}. We compute the correlation with the sum of red and green intensities (1) since the red lasers that we used during our experiments produced a strong response not only in the red but also in the green color channel of all cameras that we tested. Since we are dealing with static scenes and a fixed camera, the difference computation is a simple but very effective technique to stabilize the subsequent laser point detection.

Laser points in images usually do not appear as single bright pixels but rather as circular or oval regions several pixels in diameter with the intensity maximum in the middle and intensity quickly decreasing towards the boundary of the region. Since such an intensity distribution resembles a two-dimensional Gaussian quite well, we chose to detect laser points by computing the cross-correlation of a Gaussian and the difference image. More formally, we compute the cross-correlation of the intensity channel $f(u,v)$ with the Gaussian

$$g(u,v) := \exp\big(-(u^2+v^2)/(0.5\cdot m^2)\big), \quad (2)$$

centered at each pixel position (u,v). Here $m = 11$ is our default choice for the width and height of the cross-correlation patch size. We then detect the positions of a set of local cross-correlation maxima equal to the number of laser pointers and store them for further refinement. A simple non-maximum suppression around each detected point avoids false positives caused by nearby correlation maxima.

The laser point positions recovered as maxima of the cross-correlation function have pixel precision only. To refine the points we compute the weighted centroid over the patch of size $m \times m$, where $f(u,v)$ is used as weight. That is, for a laser point position $p = (p_x, p_y)$ the refined position p' is computed as

$$p' = \frac{1}{w}\sum_{(u,v)\in A} f(u,v)\cdot(u,v), \quad \text{with} \quad w = \sum_{(u,v)\in A} f(u,v) \quad (3)$$

where $A = [p_x - \frac{m}{2}, p_x + \frac{m}{2}] \times [p_y - \frac{m}{2}, p_y + \frac{m}{2}]$ is the patch of size m around p.

Laser points are tracked over the image sequence using the correspondence method of [Scott and Longuet-Higgins 1991] which is briefly summarized in the following. Given the sets of detected laser points $P_j := \{p_{s,j}\}$ in image j and $P_{j+1} := \{p_{t,j+1}\}$ in image $j+1$ with $m := |P_j|$, $n := |P_{j+1}|$, a distance matrix $D \in \mathbb{R}^{m\times n}$ is computed with elements

$$D_{s,t} = \exp(-||p_{s,j} - p_{t,j+1}||^2/2\sigma^2). \quad (4)$$

A value of $\sigma = 20$ has proven to work well in all our experiments. Then the singular value decomposition $D = U\Sigma V^T$ is computed, the singular values on the diagonal of Σ are all set to 1 to obtain Σ', and the matrices are multiplied back as $D' = U\Sigma' V^T$. Now a maximum in the (s,t)-th element of D' indicates that point $p_{s,j}$ best corresponds to $p_{t,j+1}$. All correspondences are detected by iteratively finding the maximal element and canceling out the respective column and row in the matrix D'. For details, see [Scott and Longuet-Higgins 1991].

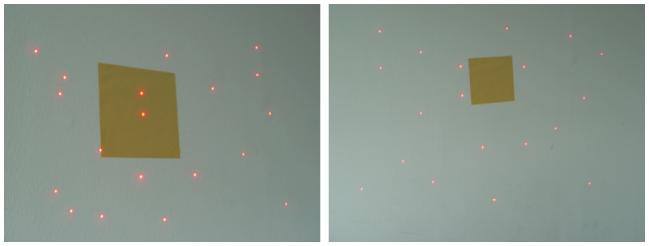

Figure 2: *Example frames of a calibration sequence. The laser rig is attached to the camera and moved in front of a wall. The square calibration pattern is used to obtain estimates of the camera's intrinsic parameters, the plane parameters, and to perform the metric upgrade after bundle adjustment.*

5 Calibration

To be able to perform depth measurements the camera needs to be calibrated and the constellation of rays in the laser rig needs to be determined. That is, we need to recover the line equation L_i for each laser ray with respect to some (arbitrary but fixed) world coordinate frame. Note that we do not need to calibrate the pose (i.e., position and orientation) of the laser rig a-priori but rather recover this information later from the images for the actual surface reconstruction (cf. Section 6). We chose the following calibration approach that allows for the simultaneous recovery of all required parameters.

The central idea is to fix the position of the camera in the origin of the world coordinate frame as well as the positions and orientations of all laser rays, and to observe the intersections of the laser rays with a varying plane. In practice, we rigidly attach the laser rig to the camera and move it in front of a wall. Additionally, we stick a colored, square piece of paper to the wall which can easily be detected (cf. Figure 2 for example frames of a calibration image sequence). In a first step we then recover estimates of the plane parameters $N_j \in \mathbb{R}^4$ (one plane for each frame, recall that j indexes frames) and the camera's intrinsic parameters $K \in \mathbb{R}^{3\times 3}$ from the detected corners of the square pattern using the method of [Zhang 2000]. Given the planes, the intrinsic parameters and the detected laser points $p_{i,j}$ in the image sequence, we compute estimates of the laser rays L_i. Note that, since the laser rig is mounted on the camera, each laser ray together with the camera center defines an epipolar plane in space. The laser points of a respective ray therefore move on an epipolar line in image space, and the epipolar planes can be recovered by determining and back-projecting these lines. Hence each laser ray can be parameterized by only 2 parameters in its epipolar plane instead of the 4 parameters required

by an unconstrained line in space. Once the estimates of the lines and planes have been computed, we refine the involved parameters by a global bundle adjustment optimization procedure. That is, we minimize the sum of squared Euclidean image space distances

$$E(L_i, N_j) = \sum_{i,j} ||p_{i,j} - K \circ q_{i,j}||^2, \quad \text{with} \quad q_{i,j} := N_j \cap L_i \quad (5)$$

being the intersection of plane N_j with laser ray L_i, and $K \circ q_{i,j}$ being the projection of the intersection into image space.

Since the optimization (5) does not constrain the world frame and especially may move the plane at infinity (see, e.g., [Hartley and Zisserman 2003] for details) to an arbitrary position, we employ the stratified metric upgrade described in [Hartley and Zisserman 2003, Chap. 10.4]. In the first step (the step from projective to affine space), the plane at infinity is moved to its canonical position. This is achieved by computing a transformation that moves the intersections of parallel lines in space to infinity. In our case the square calibration pattern provides two pairs of parallel edges per frame. The lines in space are generated by transferring the corners of the pattern to the corresponding plane in space. In the second step (the step from affine to metric space), the angles at the four corners of the pattern are adjusted to be 90 degrees.

6 Depth Map Recovery

The depth map recovery consists of two major components. The first is the estimation of the laser rig's pose discussed in Section 6.1, the second is the generation and maintenance of a laser ray to image point mapping presented in Section 6.2.

6.1 Laser Rig Pose Estimation

Suppose we are given a set of laser rays L_i in 3-space and a set of corresponding image positions $p_i \in \mathbb{R}^2$. Our goal is to find a transformation consisting of a rotation and translation that moves the laser rig to a pose such that the distance between the laser rays, projected into image space, and the corresponding laser points is minimized.

More formally, we minimize the objective function

$$E(T) := \sum_i \text{dist}(K \circ T \circ L_i, p_i)^2 \quad (6)$$

where $T \circ L_i$ denotes the transformation of a laser ray in object space, $K \circ T \circ L_i$ denotes the projection of a transformed ray into image space and $\text{dist}(\cdot, \cdot)$ denotes the Euclidean distance between a line and a point in image space. We employ an image space distance measure rather than measuring the distance between laser rays and viewing rays in object space since the latter measure is not invariant to the object-to-camera distance: The same amount of image point detection error would have a stronger influence on a laser ray far away from the camera than on a nearby ray. The transformation T is minimally parameterized by 3 quaternion parameters for rotation and 3 parameters for translation. We apply a standard Levenberg-Marquardt minimization algorithm to find the best fitting pose of the laser rig. Since the pose of the rig changes only slightly between frames of the input sequence, we use the transformation of the previous frame to initialize the optimization for the current frame. For the first frame we found that an initialization with the identity transformation (i.e., no rotation and no translation) is completely sufficient. Once the optimal transformation T has been found, the depth values for the points p_i are computed by determining the point on the viewing ray $K^{-1}p_i$ with the closest distance to the corresponding laser ray L_i.

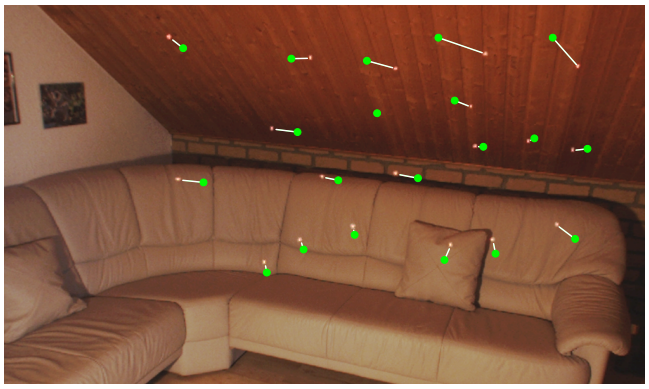

Figure 3: *An example for the result of the greedy pairing algorithm. The green dots denote the points of the best fitting pattern $\{o_i\}^*$, generated by intersecting the laser rays with a plane orthogonal to the z-axis and adjusting its scale and position. The white lines point towards the nearest detected laser points.*

The geometry of the whole scene is recovered by accumulating the per-image depth values of the entire input sequence. We then compute the Delaunay triangulation in image space and transfer it to 3-space by back-projecting all image points and moving them to their respective depths.

6.2 Laser Ray to Image Point Mapping

A consistent mapping that assigns laser rays to detected image points is the major prerequisite for the above optimization algorithm to converge to the correct pose of the laser rig. We employ two different techniques to recover this mapping for the first frame of the input sequence and for successive frames.

Finding the mapping between laser rays and image points without any a-priori knowledge is quite involved. In our experiments we use 20 laser pointers which means that there are $20! \approx 10^{18}$ possible mappings in general. Since a complete testing of all mappings is impossible, we use a simple yet effective pattern matching heuristic that worked well in our experiments if we ensure that all laser points are visible in the first frame. Given a set of detected laser points $\{p_i\}$, the idea is to generate a sequence of patterns in the form of point sets $\{o_i\}_k$ with $o_i \in \mathbb{R}^2$ by intersecting the laser rays L_i with a set of planes at varying positions. The pattern that best matches the p_i then induces the sought initial mapping. The matching quality of a particular pattern $\{o_i\}$ is defined as the sum of squared distances after greedily paring the sets $\{p_i\}$ and $\{o_i\}$ based on shortest Euclidean distance.

The sequence of patterns $\{o_i\}_k$ is constructed by intersecting the laser rays L_i with a plane perpendicular to the z-axis, positioned at varying depths in front of the laser rig, and by then rotating the intersection points around the z-axis by varying angles. The intuition behind this approach is to approximate the shape of the laser points as they would appear for scenes of different depths and different rotations of the laser rig. Instead of projecting the intersection points to image space we merely drop the third coordinate to obtain the $o_i \in \mathbb{R}^2$. Different positions of the sets $\{p_i\}$ and $\{o_i\}$ are compensated for by moving the centers of gravity of both sets to the origin. Both sets are furthermore scaled to $[-1, 1]^2$. By independently scaling the x- and y-axes of each set we effectively compensate for perspective foreshortening of the detected laser points p_i due to slanted scene geometry. In all our experiments, we let the distance of the intersection plane vary between 0 and 5 meters in

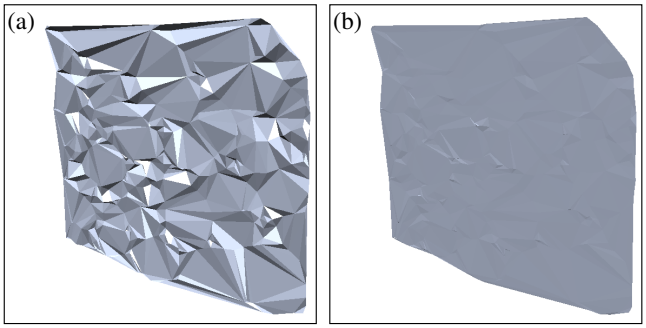

Figure 4: *Reconstruction of a plane from 20 images before (a) and after (b) application of our smoothing operator. This example clearly shows that most of the noise in the recovered depth values has a systematic character and can hence effectively be removed. Note that in (b) no per-vertex smoothing has been applied.*

steps of $10cm$ and the rotation angle between 0 and 360 degrees in steps of 10 degrees. Figure 3 shows the result of the greedy paring algorithm for the living room sequence. Green dots mark the points o_i of the winning pattern, white line segments point towards the respective laser points they have been paired with.

Once the mapping from laser rays to image points has been recovered, it can, in principle, be maintained from one frame to the next by tracking laser points. To be able to cope with occluded and later re-appearing laser points, with wrongly detected points, and with false matches, we employ a robust sampling strategy to determine the correct mapping for each frame. Given the matched features from the laser point tracker for a new frame (cf. Section 4), we generate a set of candidate laser ray to image point pairs by simply assigning all tracked points to the laser rays they corresponded to in the previous frame.

To find the correct pose of the laser rig even for unreliable laser ray to image point mappings we run a RANSAC [Fischler and Bolles 1981] based sampling algorithm that generates a set of hypotheses for the rig pose, evaluates them, and then keeps the winning hypothesis as new pose. A hypothesis is generated by randomly selecting 6 pairs from the set of candidates and by solving the minimization problem (6) for the rig pose. To evaluate a hypothesis we first greedily pair all laser rays and points *not* used to compute the hypothesis, again based on Euclidean distance in image space, and then compute the statistically robust error function

$$E(h) = \sum_i \log\left(1 + \mathrm{dist}(K \circ T_h \circ L_i, p_i)\right) \tag{7}$$

where L_i, p_i denote the current ray and point pairs and T_h denotes the hypothetical laser rig transformation. The winning hypothesis T_h^*, i.e., the one with the smallest error (7), is refined by again solving (6), this time with all laser to point pairs (L_i, p_i) that lie sufficiently close to each other, i.e., with $\mathrm{dist}(K \circ T_h^* \circ L_i, p_i) < d_{\mathrm{thresh}}$. In our experiments we found that a threshold $d_{\mathrm{thresh}} = 1$ pixel works well. We furthermore exclude pairs (L_i, p_i) that are ambiguous, i.e., pairs for which more then one image point lies within the tolerance d_{thresh} to $K \circ T_h^* \circ L_i$.

7 Depth Map Smoothing and Outlier Rejection

Apart from the standard measurement noise common to all laser-based reconstruction systems, we encountered an additional type of noise during our experiments, caused by our specific approach: The optimization procedure (6) compensates for sub-pixel laser point detection errors by determining a slightly wrong laser rig pose, thereby adding noise to the resulting depth values. However, since the orientation of the rig is rather resilient to small-scale detection errors, the error compensation mostly moves the position of the rig towards or away from the camera, i.e., introduces a systematic per-frame depth error. In other words, the depth values of a respective frame are all affected by the same offset in roughly the same direction. This observation is verified by the following experiment. We applied our method to 20 frames of an image sequence showing only a single plane and reconstructed a 3D mesh from the image space Delaunay triangulation as described above. We then computed an individual least squares plane for the vertices recovered from each single frame. The average Euclidean distance of the vertices to their respective plane was below 1.05mm, the maximal distance over all frames was 3.72mm. For comparison we then computed a single least squares plane for *all* reconstructed vertices. Now the average distance was 5.49mm and the maximal distance 16.89mm. This shows that, while the overall reconstruction (cf. Figure 4a) is affected by a considerable amount of noise, the depth values recovered in each individual frame are much more coherent. Moreover, when we compare the normal vectors of the per frame least squares planes, we see only very little variation. This indicates that the per frame groups of samples are mostly shifted by a constant depth offset.

To remove noise of the above kind we devised a simple, iterative smoothing technique for the reconstructed surface mesh. For each frame j, i.e., for all vertices $x_{i,j} \in \mathbb{R}^3$ corresponding to the same laser rig pose, our goal is to find one common update vector $m_j \in \mathbb{R}^3$ in order to improve simultaneously the *local smoothness* around all respective $x_{i,j}$. (We will drop the index j in the following to simplify the notation.) The vertices are then updated as

$$x_i \leftarrow x_i + (m^T v_i) v_i \quad \text{with} \quad v_i := K^{-1} p_i \tag{8}$$

being the viewing ray where x_i lies on. Local smoothness at a vertex x_i is measured using the length of the Laplace vector

$$l_i := \frac{1}{\Omega} \sum_{q_j \in N(x_i)} \omega_j (q_j - x_i), \qquad \Omega := \sum_j \omega_j \tag{9}$$

in the surface mesh generated from the Delaunay triangulation of the reconstructed samples from all frames. Here $N(x_i)$ denotes the set of 1-ring neighborhood vertices around x_i, i.e., all vertices that are connected to x_i by an edge in the triangulation no matter from which frame j they have been reconstructed. For the ω_i we use the cot-weights described, e.g., in [Pinkall and Polthier 1993]. Since we seek an update m which simultaneously improves the smoothness at all vertices x_i of a respective frame, we find m as the solution of the minimization problem

$$E(m) := \sum_i ||l_i||^2. \tag{10}$$

That is, for the vertices x_i recovered from a respective frame, we compute the update vector m that minimizes the sum of squared Euclidean lengths of the corresponding Laplace vectors l_i. One smoothing step of the whole scene mesh then consists of first computing an update m_j for each frame j and then applying all updates to the respective vertices. Figure 4b shows the result after the application of several smoothing iterations to the mesh of the plane example from above. Figures 5b and 5c demonstrate the effect of the smoothing operator on a real-world reconstruction example. Although the mesh before smoothing may be affected by strong noise as shown in Figure 5b, it still contains the true surface information encoded in the form of per-frame vertex configurations. Since

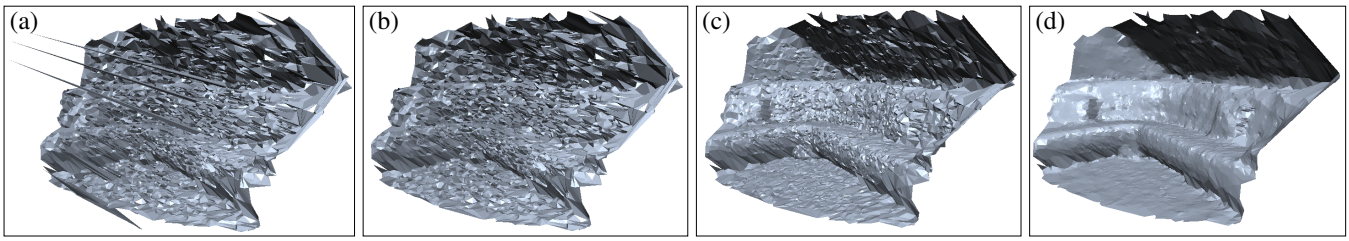

Figure 5: *Renderings of the different steps of our outlier rejection and smoothing pipeline. (a) shows the raw reconstructed surface mesh (i.e., without any post-processing) of the living room sequence. Outliers are removed in (b), and (c) shows the result of several iterations of the smoothing operator discussed in Section 7. The final result in (d) has been obtained by several additional iterations of per-vertex Laplacian smoothing where the vertices are still constrained to lie on their respective viewing ray. Even though (a) looks very noisy, the noise has a special systematic structure (cf. Section 7) and can therefore be eliminated quite effectively.*

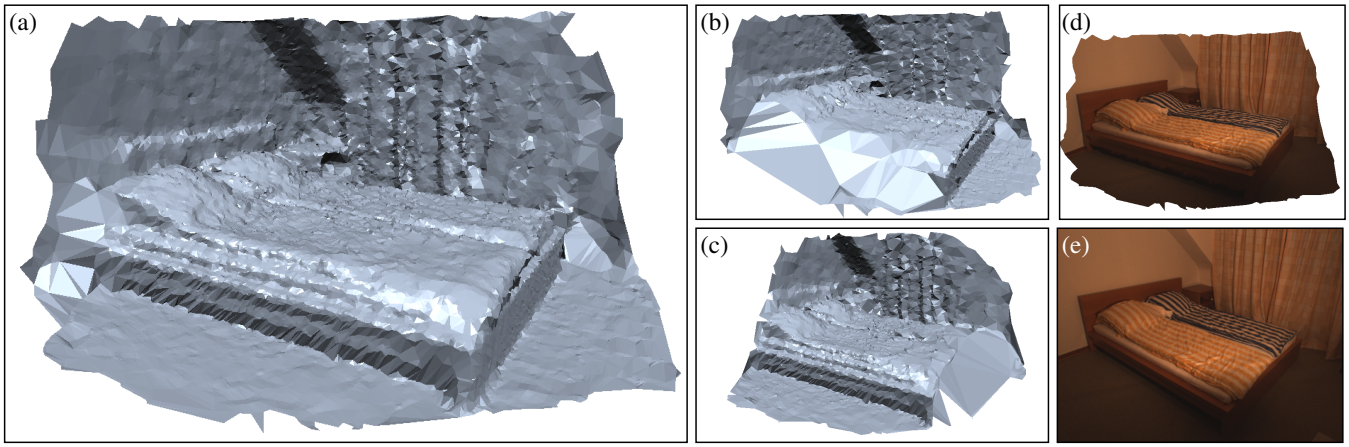

Figure 6: *Reconstruction of a bedroom scene. This example exploits the fact that the pose of the laser rig can be chosen arbitrarily: The resulting surface in (a) is a combination of two sub-sequences with different laser rig poses. The individual reconstructions of the sub-sequences are shown in (b) and (c), respectively. (d) shows a texture-mapped rendering and (e) one of the input images.*

Resolution	720×576	1024×768	1280×960
Avg. time	119ms	205ms	333ms

Table 1: *Average per-image computation time required for the detection of 20 laser points in images of different resolutions using our cross-correlation based detection algorithm.*

these per-frame configurations are moved rigidly by the smoothing algorithm, our method is able to faithfully recover the correct surface up to inevitable measurement noise. The final step of the mesh post-processing (cf. 5d) hence consists of a standard Laplacian smoothing operator that still constrains all vertices $x_{i,j}$ to lie on their respective viewing ray.

Another issue we have to deal with are outlying depth values, which are the result of wrong laser ray to image point pairs. Wrong pairs may be generated during the greedy pairing process if a false image point is closer to a projected laser ray than the correct point either since the correct point is occluded or due to slight detection errors. Outlying pairs do not harm the laser rig pose estimation: they are only generated since they are supported by the winning hypothesis for the rig pose and hence do not influence the final pose refinement (6) much because outliers with $\mathrm{dist}(\cdot,\cdot) > d_{\mathrm{thresh}}$ are not taken into account. They do, however, usually appear as long, thin spikes in the recovered scene geometry (cf. Figure 5). We implemented a simple outlier detection mechanism based on the length of object space edges and on the length ratio of object space to image space edges of the generated Delaunay triangulation. All faces that contain an edge longer then l_{thresh} or with a ratio larger than r_{thresh} are erased. After that we erase all connected mesh components that only consist of 10 or less vertices. In our experiments we found that thresholds of $l_{\mathrm{thresh}} = 50cm$ and $r_{\mathrm{thresh}} := 4$ work sufficiently well. Outlying vertices are removed by re-computing the image space Delaunay triangulation without them. The effect of the outlier removal is demonstrated in Figure 5a for the living room image sequence.

8 Results

All results and measurements presented in the following have been performed on an AMD Athlon64 based system running at 2.2GHz. The computation time required by the laser point detector is shown in Table 1 for several different image resolutions. The laser point tracking algorithm takes, on average, 1.59ms per frame for 20 laser points and is hence negligible in comparison to the time required for the laser point detection. The time required for the estimation of the laser rig pose strongly depends on the quality of the initialization, i.e., on the distance between the initial and the optimal pose. In combination with the RANSAC approach, this means that the required computation time for the hypotheses depends on how many false laser ray to image point pairs they contain. For the living room scene (see below) the per-frame estimation (including the random sampling and the final refinement) took between 1574ms and 90ms,

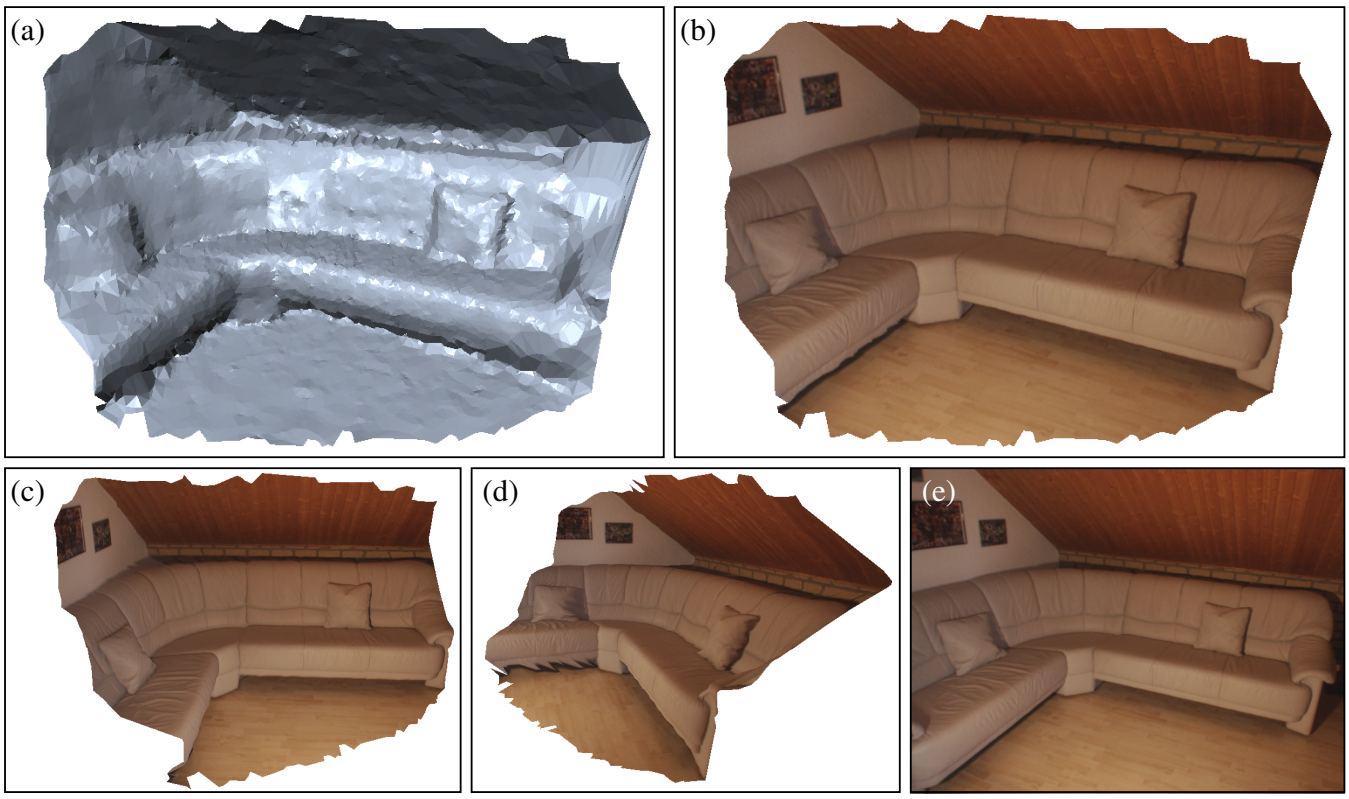

Figure 7: *Partial reconstruction of a living room scene. The recovered triangle mesh is shown as flat shading in (a) and texture mapped in (b). (c) and (d) show textured renderings from new vantage points and (e) shows one of the input images.*

with an average of 135ms. The long maximal estimation time is caused by the initialization of the laser rig's pose transformation in the first frame: The identity transformation is, in general, quite far away from the optimal transformation. Hence the solution of the optimization problem (6) requires much more time than for successive frames where the previous transformation is used for initialization. The generation of the Delaunay triangulation and the post-processing, i.e., smoothing and outlier detection, for all examples always took less than one minute.

An issue requiring consideration is the camera setup, especially the shutter speed. Clearly, if the exposure time is too long, the projections of laser points in the scene result in short lines segments in the input images rather than points. It hence is necessary to find a compromise between too long exposure times and too dark input images. Since the laser points are usually very bright in regularly illuminated indoor scenes, we found that it is possible to set the exposure time to small values and still be able to robustly detect laser points. Concretely, we set the shutter speed to 20ms–12ms (i.e., $1/50$s–$1/80$s) in all our experiments. These values allow for laser projections without motion blur if the laser rig is moved at moderate speeds.

In the first example in Figure 6 we have applied our method to a bedroom scene. The main difficulty here is the non-trivial geometry of the curtain and the wrinkles in the blanket that cause laser points to be occluded and re-appear frequently. Nevertheless our algorithm is able to find a sufficient number of correct laser ray and image point pairs to perform the reconstruction. In this example we have utilized the fact that the user may choose the pose of the laser rig freely. The complete geometry has been recovered from two subsequences, one with the laser rig on the left side of the camera and one with the rig on the right side. The two partial reconstructions (shown in Figures 6b and 6c, respectively) have been integrated into a single surface by combining the image-space Delaunay triangulations. Outlier removal and smoothing has been performed on the combined surface to exploit the increased resolution of the resulting mesh. Note that the rough appearance of the curtain in the back of the scene is due to triangulation artifacts caused by the sparse sampling. Better results might be obtained by using a data-dependent triangulation instead of a simple Delaunay triangulation. The input image sequence consisted of 1350 images of resolution 1024×768 in total and the reconstruction took 10.5 minutes, including image loading, point detection and matching, and rig pose estimation. The final mesh consists of 21.5k vertices. Figure 6d shows a texture-mapped rendering of the reconstructed mesh from a new vantage point. For texturing we reuse the image I_{empty} of the difference computation in Section 4, i.e., an image taken from the same camera position as the images of the reconstruction phase but without laser points.

The next example (cf. Figure 7) shows the reconstruction of a living room scene. We took 630 images (resolution 1024×768) of this scene, the overall computation then took slightly more than 4 minutes. The final triangle mesh consists of 10k vertices.

The third example demonstrates the robustness of our algorithm with respect to falsely detected laser points. The specular white tiles (cf. Figure 8b) reflect the laser rays to the other wall, where they produce additional strong local maxima in the cross-correlation function of the laser point detector. These false positives then easily overrule dull laser points caused by less reflective surface material. However, the robust sampling approach enables our algorithm to cope well with this situation and the only possible side-effect is, as

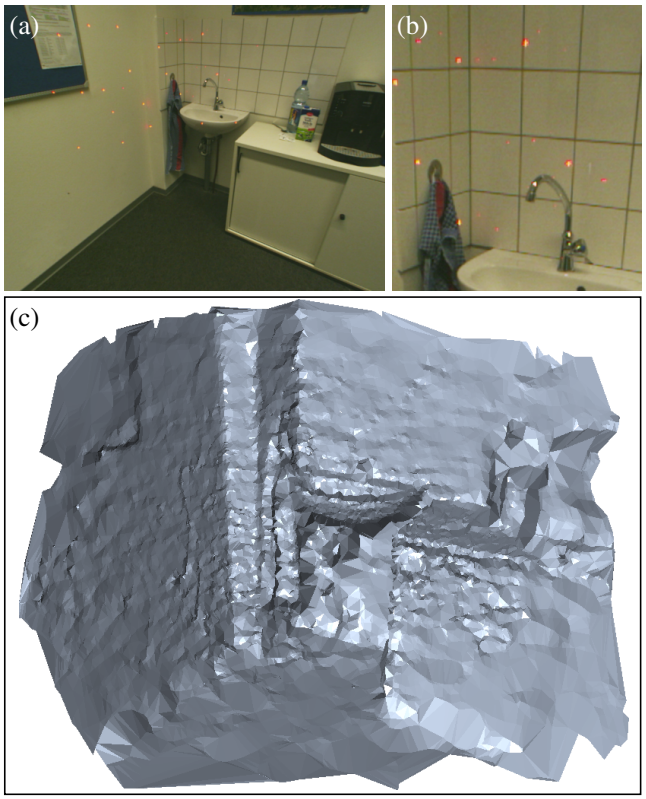

Figure 8: *Reconstruction of a scene with strongly reflecting surface parts. (a) shows one frame of the input image sequence, (b) shows a closeup of the tiled surface with several additional red dots caused by reflections. Note that the two bright reflections on the water tap result from the ceiling light, not from laser pointers. The reconstructed triangle mesh is shown in (c).*

discussed above, outlying vertices in the final surface which can effectively be detected and removed. For this example we used 900 input images of resolution 1280×960, resulting in a triangle mesh of 13.4k vertices after a total computation time of 9 minutes.

9 Conclusion and Discussion

We have presented a new method for the incremental reconstruction of depth maps using a hand-held array of laser pointers. The main advantages of the proposed approach are its flexibility, its versatility and its robustness: The pose and motion of the laser rig is completely unconstrained, allowing the user to move the lasers freely through the scene and enabling a variety of possible reconstruction setups. We have demonstrated with several example reconstructions of scenes with non-trivial geometry and surface materials that our algorithm robustly copes with occluded, falsely detected or falsely matched laser points. Nevertheless, some of the problems inherent to other laser-based reconstruction techniques apply to our method as well: completely transparent or mirroring materials (cf. the water tap in Figure 8b) or materials that completely absorb the laser light cannot be reconstructed. Furthermore and again similar to existing methods, the more parallel the laser rays are to the viewing rays of the camera, the more severe becomes measurement noise which cannot be removed with the smoothing operator presented in Section 7. Hence our method works best for a large camera viewing frustum and sufficiently large angles between viewing and laser rays.

We have identified several areas of possible future work. On the one hand we plan to turn our method into a real-time system that provides the user with instant feedback on where the scene has been sufficiently covered with laser points already and which parts require more coverage with laser points. Furthermore, our current approach of computing the Delaunay triangulation in image space in combination with sparse samplings may result in suboptimal surfaces. We plan to investigate data-dependent triangulation approaches like the one by [Dyn et al. 1990] to counter these surface artifacts. To overcome the problem of sparse image space samples altogether we plan to add a laser plane emitter to the laser rig. This way we could recover a whole image space line of samples per frame in addition to the sparse set of samples from the laser pointers.

Acknowledgments

This project was funded by the DFG research cluster "Ultra High-Speed Mobile Information and Communiation" (UMIC), http://www.umic.rwth-aachen.de/.

References

BAHMUTOV, G., POPESCU, V., AND MUDURE, M. 2006. Efficient large-scale acquisition of building interiors. *Computer Graphics Forum 25*.

BESL, P. J. 1989. Active optical range imaging sensors. *Advances in Machine Vision*.

BLAIS, F. 2004. Review of 20 years range sensor development. *Journal of Electronic Imaging 13*, 1.

BOUGUET, J.-Y., AND PERONA, P. 1998. 3d photography on your desk. 43–50.

BUHMANN, J. M., FELLNER, D. W., HELD, M., KETTERER, J., AND PUZICHA, J. 1998. Dithered color quantization. C219–C231.

DYN, N., LEVIN, D., AND RIPPA, S. 1990. Data dependent triangulations for piecewise linear interpolation. *IMA Journal on Numerical Analysis 10*, 1, 137–154.

FELLNER, D. W., AND HELMBERG, C. 1993. Robust rendering of general ellipses and elliptical arcs. *ACM Trans. Gr. 12*, 3 (July), 251–276.

FISCHLER, M. A., AND BOLLES, R. C. 1981. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM 24*, 6, 381–395.

FOLEY, J. D., VAN DAM, A., FEINER, S. K., HUGHES, J. F., AND PHILLIPS, R. 1993. *Introduction to Computer Graphics*. Addison-Wesley.

FURUKAWA, R., AND KAWASAKI, H. 2003. Interactive shape acquisition using marker attached laser projector. In *Proc. of 3DIM*, 491–498.

HARTLEY, R., AND ZISSERMAN, A. 2003. *Multiple View Geometry in Computer Vision*, second ed. Cambridge University Press.

KAWASAKI, H., FURUKAWA, R., AND NAKAMURA, Y. 2006. 3d acquisition system using uncalibrated line-laser projector. In *Proc. of ICPR*, 1071–1075.

KOBBELT, L., STAMMINGER, M., AND SEIDEL, H.-P. 1997. Using subdivision on hierarchical data to reconstruct radiosity distribution. C347–C355.

LAFORTUNE, E. P., FOO, S.-C., TORRANCE, K. E., AND GREENBERG, D. P. 1997. Non-linear approximation of reflectance functions. In *Proc. SIGGRAPH '97*, vol. 31, 117–126.

LOUS, Y. L. 1990. Report on the First Eurographics Workshop on Visualization in Scientific Computing. 371–372.

PINKALL, U., AND POLTHIER, K. 1993. Computing discrete minimal surfaces and their conjugates. *Experimental Mathematics 2*, 15–36.

PIPITONE, F., AND HARTLEY, R. 2006. A structured light range imaging system using a moving correlation code. In *Proc. of 3DPVT*, 931–937.

POPESCU, V., SACKS, E., AND BAHMUTOV, G. 2004. Interactive modeling from dense color and sparse depth. In *Proc. of 3DPVT*, 430–437.

POPESCU, V., BAHMUTOV, G., MUDURE, M., AND SACKS, E. 2006. The modelcamera. *Graphical Models 68*, 385–401.

ROCCHINI, C., CIGNONI, P., MONTANI, C., PINGI, P., AND SCOPIGNO, R. 2001. A low cost 3d scanner based on structured light. *Computer Graphics Forum 20*, 3, 299–308.

SCHARSTEIN, D., AND SZELISKI, R. 2002. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *IJCV 47*, 7–42.

SCHARSTEIN, D., AND SZELISKI, R. 2003. High-accuracy stereo depth maps using structured light. In *Proc. of CVPR*, 195–202.

SCOTT, G. L., AND LONGUET-HIGGINS, H. C. 1991. An algorithm for associating the features of two images. In *Proc. Royal Society London*, vol. 244, 21–26.

SEIDEL, H.-P. 1993. Polar forms for geometrically continuous spline curves of arbitrary degree. *ACM Trans. Gr. 12*, 1 (Jan.), 1–34.

SEITZ, S., CURLESS, B., DIEBEL, J., SCHARSTEIN, D., AND SZELISKI, R. 2006. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *Proc. of CVPR*, vol. 1, 519–526.

TAKATSUKA, M., WEST, G. A. W., VENKATESH, S., AND CAELLI, T. M. 1999. Low-cost interactive active monocular range finder. In *Proc. of CVPR*, 444–449.

WEISE, T., LEIBE, B., AND GOOL, L. V. 2007. Fast 3d scanning with automatic motion compensation. In *Proc. of CVPR*.

WINKELBACH, S., MOLKENSTRUCK, S., AND WAHL, F. 2006. Low-cost laser range scanner and fast surface registration approach. In *Pattern Recognition (DAGM 2006), Lecture Notes in Computer Science 4174*. Springer, 718–728.

ZAGORCHEV, L., AND GOSHTASBY, A. 2006. A paintbrush laser range scanner. *Computer Vision and Image Understanding 101*, 65–85.

ZHANG, Z. 2000. A flexible new technique for camera calibration. *IEEE PAMI 22*, 11, 1330–1334.