

Low-Cost Laser Range Scanner and Fast Surface Registration Approach

Simon Winkelbach, Sven Molkenstruck, and Friedrich M. Wahl

Institute for Robotics and Process Control, Technical University of Braunschweig,
Mühlenpfordtstr. 23, D-38106 Braunschweig, Germany
{S.Winkelbach, S.Molkenstuck, F.Wahl}@tu-bs.de

Abstract. In the last twenty years many approaches for contact-free measurement techniques for object surfaces and approaches for 3d object reconstruction have been proposed; but often they still require complex and expensive equipment. Not least due to the rapidly increasing number of efficient 3d hard- and software system components, alternative low-cost solutions are in great demand. We propose such a low-cost system for 3d data acquisition and fast pairwise surface registration. The only hardware requirements are a simple commercial hand-held laser and a standard grayscale camera.

1 Introduction

Triangulation-based laser range finders and light-striping techniques are well-known since more than twenty years (e.g. [1], [2]). Beside other active techniques – like structured light, coded light, time of flight, Moiré interferometry, etc. (see e.g. [3] for an overview) – laser range scanners are commonly used for contactless measuring of surfaces and 3d scenes in a wide range of applications. The field of application comprises computer graphics, robotics, industrial design, medical diagnosis, archaeology, multimedia and web design, as well as rapid prototyping and computer-aided quality control. Most commercial laser scan systems use a camera and a laser beam or laser plane. The surface recovery is based on triangulation, i.e. the intersection of the illuminating laser beam and the rays projected back to the camera. Expensive high-precision actuators are often used for rotating/translating the laser plane or for rotating/translating the object.

Some alternative hand-held devices avoid expensive actuators and furthermore improve the flexibility of the scanning process. These approaches have to determine the position and orientation of the laser device on-line. Such an on-line tracking is done by various mechanisms like optical LED tracking, electromagnetic sensors or mechanical positioning arms (see e.g. [3], [4]).

Instead of an external tracking system, we propose a real-time self-calibration of a hand-held laser plane, which is based on a simple analysis of the laser stripes in the camera images. Thus, the laser line can be swept manually over the object during the scan, which has several advantages: (i) Only the lightweight laser has to be held, which allows a convenient scanning process. (ii) The low-cost

hardware requirements are even affordable for students and novice developers. (iii) The illumination direction is flexible and allows an interactive avoidance of laser shadow problems and outliers. The only precondition is a known background geometry, which serves as laser calibration target.

Thus, our approach can be regarded as a generalization of Zagorchev and Goshtasby [4]. They use a reference double-frame, which is placed around the object and acts as calibration target. In their approach, the laser is calibrated using the four visual intersection points of the laser and the double-frame. Detection of the red laser lines relies on an appropriate analysis of the red components in color images. However, a standard color camera may impair the detection accuracy, since only every fourth pixel can capture red light, the remaining pixels are interpolated. Our approach exhibits several advantages over [4]: A precise reference double-frame, which has to be adapted to the object size, is not needed. The calibration target can be almost arbitrarily shaped (e.g. an arbitrary background). Moreover, the laser calibration is more robust and accurate, since we use much more than four points to calibrate the laser, and we rely on subpixel analysis of grayscale difference images. Due to our subsequent fast surface registration, the object can be moved freely between different scans. Thus, a repositioning and recalibration of the camera to get different viewing directions is not necessary; it is easily possible to scan an object from *all* sides, even from the bottom.

An outline of the numerous publications dealing with registration techniques would go beyond the scope of this paper. Therefore, we only give a short overview of the most related work: A very popular surface registration approach is the *iterative closest point* (ICP) algorithm from Besl and McKay [5]. The algorithm iteratively improves an initial solution according to some fitness criterion. Although many enhancements to the original method have been suggested (e.g. [6], [7]), it still requires a good initial guess to find the global optimum. Most approaches are using surface features to find corresponding point pairs. Features vary from simple properties like *curvatures*, to complex vectors like *point signatures* [8], *surface curves* e.g. [9], [10], *spin-images* [11] or *salient points* [12]. However, their usage cannot guarantee unique point correspondences; nevertheless, it can highly constrain the search space. A well-known category dealing with object recognition and localization are the *pose clustering* approaches (also known as *hypothesis accumulation* or *generalized Hough transform* e.g. [13]). The drawback of voting tables is their high time and space complexity, particularly in case of large data sets and high-dimensional search spaces.

The authors of [14] give an thorough overview of current registration techniques and propose a new approach, which is based on a genetic algorithm to find pose hypotheses and a novel *surface interpenetration measure* as quality criterion. Unfortunately, this complex approach needs triangle meshes and a substantial run-time (5 minutes for 10000 points on a 1.7 GHz PC for a pairwise match). We will show that our approach is able to achieve an adequate match of bigger data sets in less than a second.

The surface registration method is a significant improvement of the *random sample matching* [15], which is an efficient and robust approach for matching

fragments of broken objects without knowing an initial solution. This method is based on the *RANSAC* algorithm introduced in [16]. The repeated procedure is simple but powerful: First, a likely hypothesis is generated randomly from the input data set. Subsequently, the quality of the hypothesis (number of contact points) is evaluated.

2 Hand-Held Laser Scanner

The basic idea of our self-calibrating laser scanner is quite simple. The laser ray, expanded to a plane by a cylindrical lens, has to intersect two things at the same time: the (unknown) surface, and the a priori known reference geometry (usually the background). The visible intersection with the background is used to calibrate the laser, i.e. to calculate the exact 3d pose of the laser plane E_{Laser} . With this knowledge we can triangulate new 3d point coordinates of the object's surface by intersecting the laser plane with the projecting rays. Certainly, the camera must have been calibrated so that its external and internal parameters are exactly known. In our setup we use markers on the background and Tsai's camera calibration method [17]. Thus, the exact coordinates of the background structure with respect to the camera coordinate system are implicitly known.

2.1 On-line Laser Calibration

In many cases, a simple background (e.g. the natural environment) can be used for laser calibration. Under the assumption that the background geometry is known, we can obtain some point coordinates of the visible laser line by intersecting background and camera projection rays. Provided that these 3d intersection points are linearly independent, they constrain all degrees of the laser plane's pose. Although many background shapes are imaginable, the probably most applicable, available, and easy-to-use background will be the corner of a room, or two solid boards standing together in an exactly known angle. It

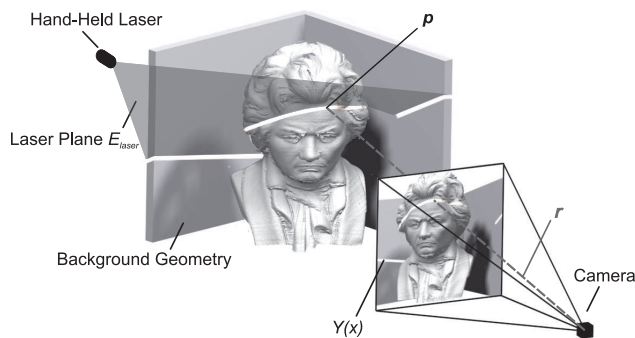


Fig. 1. Laser triangulation: 3d scene and 2d camera image. The intersection of a projection ray r with the laser plane E_{Laser} results in new 3d point p .

is important to find the coordinates of the laser line in the camera images I as precisely as possible. Thus, it is useful to take a reference image I_R without laser light and to use difference images $I_d = I - I_R$ in the following. As the laser line will be rather horizontal or vertical, we can reduce the problem to a 1d detection of the laser line in each single column or row of the image, respectively. Without loss of generality, we assume in the following that the laser line is rather horizontal. We can find the line with subpixel accuracy by calculating the (weighted) average $Y(x)$ of the “bright” pixel coordinates in each column x .

After obtaining the function $Y(x)$ of the laser line, our next task is to calculate the 3d pose of the laser plane. We use the RANSAC method [16] to repeatedly select three random pixels $Y(x_1)$, $Y(x_2)$, $Y(x_3)$, and assume that they belong to the background. Since the camera’s internal and external parameters have been calibrated, we can obtain the equation of three “light rays” r_i for each of these pixels and intersect them with the known background geometry, resulting in three surface points \mathbf{p}_1 , \mathbf{p}_2 , \mathbf{p}_3 . Unless they are linearly dependent, they define a possible laser plane pose. These hypotheses can be quickly computed and evaluated using the number of inliers of $Y(x)$ as a quality criterion.

2.2 Triangulation of 3d Points

From the previous step, we know the equation of the laser plane E_{Laser} and a number of image pixels from $Y(x)$ that are both in that plane and on the object we are scanning (see Fig. 1). Again we can obtain the equation of a “light ray” r for each of those pixels. A new surface point of our object can be easily computed by the intersection $\mathbf{p} = r \cap E_{Laser}$.

In the process of scanning, the user generally sweeps the laser plane over the object multiple times. In this way, he can “brush over” outlying values and increase the precision where necessary. Thus, it often happens that the algorithm obtains several surface points for the same image pixel. These should be merged using averaging (fast and easy) or median filtering (memory consuming and slower, but very useful w.r.t. possible outliers).

All 3d points collected with this procedure form the visible surface of the object from one viewing direction. To obtain a full 360° model, scans from different directions have to be registered.

3 Fast Surface Registration

Given a set \mathcal{P}_A of 3d point coordinates $\mathbf{p}_1, \dots, \mathbf{p}_k$ of the surface A and a set \mathcal{N}_A of corresponding 3d surface normals $\mathbf{n}_1, \dots, \mathbf{n}_k$ (outward-pointing unit vectors) at these points. Referring to [11], we call the combination of a point with its normal an *oriented point*. This gives us the set of oriented points \mathcal{A} of surface A and the set of oriented points \mathcal{B} of the counter surface B

$$\mathcal{A} := \{\mathbf{u} = [\mathbf{p}_u, \mathbf{n}_u] \mid \mathbf{p}_u \in \mathcal{P}_A \text{ and } \mathbf{n}_u \in \mathcal{N}_A\}, \quad (1)$$

$$\mathcal{B} := \{\mathbf{v} = [\mathbf{p}_v, \mathbf{n}_v] \mid \mathbf{p}_v \in \mathcal{P}_B \text{ and } \mathbf{n}_v \in \mathcal{N}_B\}. \quad (2)$$

A *tangential contact* between two oriented points $\mathbf{a} \in \mathcal{A}$ and $\mathbf{b} \in \mathcal{B}$ means that the point coordinates and the respective surface normals coincide. We say \mathbf{a} is in tangential contact with \mathbf{b} if $\mathbf{p}_a = {}^A\mathbf{T}_B \cdot \mathbf{p}_b$ and $\mathbf{n}_a = {}^A\mathbf{T}_B \cdot \mathbf{n}_b$, where ${}^A\mathbf{T}_B$ is the relative transformation in homogeneous coordinate notation. We can construct a pose hypothesis by assuming a contact between some points on each surface. More precisely, four given oriented surface points $\mathbf{a}, \mathbf{c} \in \mathcal{A}$ and $\mathbf{b}, \mathbf{d} \in \mathcal{B}$ are sufficient, if we assume a tangential contact between \mathbf{a} and \mathbf{b} as well as between \mathbf{c} and \mathbf{d} . This assumption constrains all degrees of freedom of the relative transformation. As illustrated in Fig. 2 (left), we can determine the homogeneous 4×4 transformation matrix by multiplying two frames ${}^A\mathbf{T}_B = \mathbf{F}(\mathbf{a}, \mathbf{c})^{-1} \cdot \mathbf{F}(\mathbf{b}, \mathbf{d})$, where the function $\mathbf{F}(\mathbf{u}, \mathbf{v})$ represents a coordinate system lying between the oriented points \mathbf{u} and \mathbf{v}

$$\mathbf{F}(\mathbf{u}, \mathbf{v}) := \begin{bmatrix} \frac{\mathbf{p}_{uv} \times \mathbf{n}_{uv}}{\|\mathbf{p}_{uv} \times \mathbf{n}_{uv}\|} & \mathbf{p}_{uv} & \frac{\mathbf{p}_{uv} \times \mathbf{n}_{uv} \times \mathbf{p}_{uv}}{\|\mathbf{p}_{uv} \times \mathbf{n}_{uv} \times \mathbf{p}_{uv}\|} & \frac{\mathbf{p}_u + \mathbf{p}_v}{2} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3)$$

with the difference vector $\mathbf{p}_{uv} := (\mathbf{p}_v - \mathbf{p}_u) / \|\mathbf{p}_v - \mathbf{p}_u\|$ and the combined normal vector $\mathbf{n}_{uv} := \mathbf{n}_u + \mathbf{n}_v$. To avoid singular frames, we must ensure that the length of \mathbf{p}_{uv} and \mathbf{n}_{uv} is not zero. An exact coverage of both point pairs with opposed normals is only possible if their relative distances and angles are identical. To verify this constraint, we define a 4d relation vector of an oriented point pair

$$\text{rel}(\mathbf{u}, \mathbf{v}) := \begin{bmatrix} d_{uv} \\ \cos \alpha_{uv} \\ \cos \beta_{uv} \\ \delta_{uv} \end{bmatrix} := \begin{bmatrix} \|\mathbf{p}_v - \mathbf{p}_u\| \\ \mathbf{n}_u \cdot \mathbf{p}_{uv} \\ \mathbf{n}_v \cdot \mathbf{p}_{uv} \\ \text{atan2}(\mathbf{n}_u \cdot (\mathbf{p}_{uv} \times \mathbf{n}_v), (\mathbf{n}_u \times \mathbf{p}_{uv}) \cdot (\mathbf{p}_{uv} \times \mathbf{n}_v)) \end{bmatrix}, \quad (4)$$

consisting of the Euclidean point distance d_{uv} , the angles of inclination α_{uv} and β_{uv} between the normals \mathbf{n}_u and \mathbf{n}_v , the line connecting \mathbf{p}_u and \mathbf{p}_v , and finally the rotation angle δ_{uv} between the normals around the connection line. The four relations are also illustrated in Fig. 2 (right). Note that the relation vector

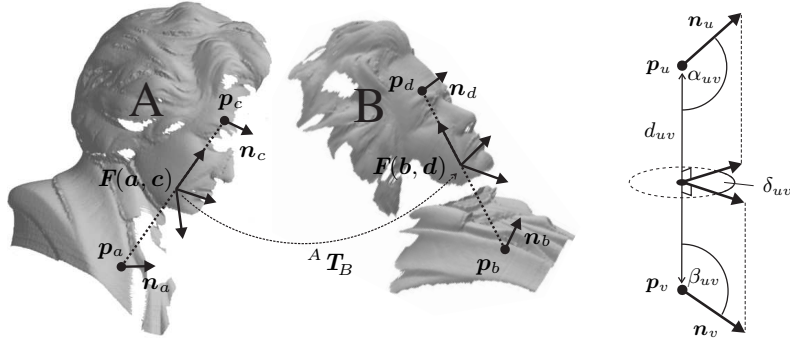


Fig. 2. (Left) relative transformation ${}^A\mathbf{T}_B$ between supposed contact points; (right) relations between the oriented points \mathbf{u} and \mathbf{v} .

is invariant under rotation and translation. Incidentally, Wahl, Hillenbrand and Hirzinger [18] showed that similar relation vectors can be accumulated in feature histograms for rapid 3d-shape classification. Using these relation vectors, the set of valid pose hypotheses \mathcal{H} can be specified by

$$\mathcal{H} := \{(\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}) \mid \text{rel}(\mathbf{a}, \mathbf{c}) = \text{rel}(\mathbf{b}, \mathbf{d}); \mathbf{a}, \mathbf{c} \in \mathcal{A}; \mathbf{b}, \mathbf{d} \in \mathcal{B}\} . \quad (5)$$

3.1 Rapid Generation of Likely Pose Hypotheses

In [15] we have proposed a highly efficient method for generating likely pose hypothesis by using 'spin-tables'. The following improved approach considerably accelerates the run-time of the hole matching algorithm. In our experiments we observe an acceleration factor of 40 to 100.

How long does it take to find two corresponding point pairs (one for A and one for B)? Assume, that we have two identical surfaces, each with n surface points. Having chosen a point pair of A, the probability to select the corresponding point pair of B by chance is $1/n^2$. Thus, we have to compare an average of $n^2 + 1$ point pairs, which results in an expensive run-time complexity of $O(n^2)$. But with a simple trick, the problem can be computed much faster:

Assume we alternately choose random point pairs of A and B, and store them in a hash table, using rotational invariants as table indices. Under the assumption that the invariants are unique, we only need to process an average of $1.2 \cdot n$ pairs until a hash collision occurs. This will provide the much better run-time complexity of $O(n)$. This approach complies with the 'birthday attack' [19] - an efficient cryptological strategy to generate two different documents with similar digital signatures (hash values). Let us concretize the algorithm. Instead of a hash table, we use 4d relation tables (one per surface), and the four invariant relations (4) as table indices. This leads to the following search loop:

1. Randomly choose an oriented point pair $\mathbf{a}, \mathbf{c} \in \mathcal{A}$ and calculate $\text{rel}(\mathbf{a}, \mathbf{c})$.
2. Insert the point pair into the relation table: $R_A[\text{rel}(\mathbf{a}, \mathbf{c})] = (\mathbf{a}, \mathbf{c})$.
3. Read out same position of the opposite relation table: $(\mathbf{b}, \mathbf{d}) = R_B[\text{rel}(\mathbf{a}, \mathbf{c})]$; if there is an entry \Rightarrow new pose hypothesis $(\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d})$.
4. Randomly choose an oriented point pair $\mathbf{b}, \mathbf{d} \in \mathcal{B}$ and calculate $\text{rel}(\mathbf{b}, \mathbf{d})$.
5. Insert the point pair into relation table: $R_B[\text{rel}(\mathbf{b}, \mathbf{d})] = (\mathbf{b}, \mathbf{d})$.
6. Read out same position of the opposite relation table: $(\mathbf{a}, \mathbf{c}) = R_A[\text{rel}(\mathbf{b}, \mathbf{d})]$; if there is an entry \Rightarrow new pose hypothesis $(\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d})$.

These steps will be repeated until the hypothesis is good enough, all combinations are tested, or the time exceeds a predefined limit. Optionally the hypotheses selection in step 3 and 6 can be improved further by comparing local features; i.e. we only select hypotheses that satisfy $feature(a) = feature(b)$ and $feature(c) = feature(d)$. In our experiments we use the local mean curvature, which enables us to reject over 95%. We found that 4d relation tables with 32^4 entries offer a good trade-off between accuracy and efficiency. Using 2×2 bytes per entry, one relation table requires a reasonable memory capacity of four megabytes. The proposed algorithm offers a run-time complexity of $O(n)$

for the first hypothesis, but since the relation tables get filled continuously, the complexity converges to $O(1)$ for further hypotheses.

3.2 Fast Hypotheses Verification

After generating a pose hypothesis we must measure its matching quality. For this we adopt the approach of [15], where the proportion of overlapping area Ω (where surface A is in contact with the opposite surface B) is estimated. We assume that the surfaces are in contact at areas where the distances between surface points are smaller than some predefined ε . In contrast to [15] we do not have to consider fragment penetrations. Suppose that $\mathbf{x}_1, \dots, \mathbf{x}_n \in A$ are independent random points. Let $contact_B(x)$ be a function which determines whether a point \mathbf{x} is in contact with surface B

$$contact_B(x) = \begin{cases} 1 & \text{if } dist_B(\mathbf{x}) < \varepsilon, \\ 0 & \text{else} \end{cases} \quad \text{with } dist_B(\mathbf{x}) = \min_{\mathbf{y} \in B} \|\mathbf{x} - {}^A\mathbf{T}_B \cdot \mathbf{y}\|. \quad (6)$$

The function $dist_B(\mathbf{x})$ returns the minimal distance of a point \mathbf{x} w.r.t. surface B . It can be implemented efficiently by using a *kd-tree* data structure (see [20]), which offers a logarithmical time complexity for the closest point search. Now Ω can be approximated up to an arbitrary level of confidence. Considering the margin of error, for every additional random point, the approximation of Ω can be recomputed as

$$\Omega \approx \frac{\sum_{i=1}^n contact_B(\mathbf{x}_i)}{n} \pm \frac{1.96}{2\sqrt{n}} \quad (7)$$

with a 95% level of confidence. Ω can also be regarded as the probability that a random point $x \in A$ is in contact with the opposite surface B . Thus Ω can be forecasted by an efficient *Monte-Carlo* strategy using a sequence of random points, combined with a dropout if the upper bound of the confidence interval is considerably worse than the last best match. In this manner the quality estimation gets faster and faster, whenever the hypothesis is improved.

4 Experimental Results and Conclusion

For experimental evaluation, we used a grayscale CCD camera with XGA resolution, connected to a standard AMD-Athlon PC with 2.2 GHz. The scanning accuracy naturally depends on the exactness of the camera calibration and on the triangulation angle. To evaluate the accuracy of our laser scanner, we have scanned a well-known test object under a reasonable triangulation angle of about 30-35°, and a distance of 600 mm to the camera. The object's front surface consists of two planar faces with a 50.25 mm step in depth. The scan result contains this step within a tolerance of less than 0.4 mm. The measured (unfiltered) depth values of each surface are very accurate and show an RMS error of only 0.37 mm. Fig. 3 shows a scan line of the unfiltered depth values. The accuracy can be further improved by using appropriate time and space filtering (average and/or median).

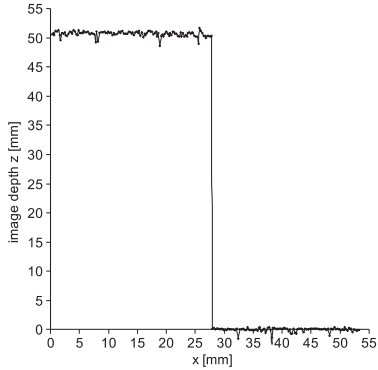


Fig. 3. Scan line of two planar faces with a 50.25mm step in depth.

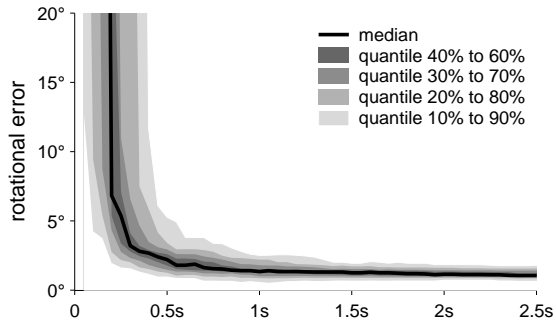


Fig. 4. Rotational registration error over time, median \pm quantiles of 100 test runs.

Our surface registration approach performs very well with all of our test objects. Since the registration algorithm is a random process, we carried out 100 test series of each pairwise match. Fig. 4 shows the median rotational error over time in the case of two scans of the Beethoven bust (shown in Fig. 2). Both surfaces consist of approximately 60000 points with a surface overlap of 35%. As can be seen, after an execution time of only 0.5 seconds, 50% of all passes have already achieved a rotational accuracy of less than 2° . For accuracy evaluation we used a high-precision turn table, which provides the 'ground truth'. The outstanding registration efficiency is also documented in Table 1. The table presents a comparative study of the root mean square (RMS) distance, rotational error, and execution time between the registration approach in [15] and our substantial improvement. As can be seen, we achieve an impressive speed-up of factor 80 in this case. The results of all other test series are similar (factor 40–100 depending on data size and overlap).

Fig. 5 presents some result images of our scanning and registration method. In the first row one can see a side view of four scans of a clay mole (height 145 mm) under different angles of rotation of about 90° from one to the next. The second row shows a camera image of the mole and three images of the registration result of all four scans. In the third image of this row, the different scans have been colored in different shades of gray in order to show the surface inter-

Table 1. Registration performance of 100 test runs: comparison of root mean square (RMS) distance, rotational error (RotE), and execution time to achieve an RMS distance of less than 3 mm

	using approach of [15]				after our improvement			
	mean	median	min	max	mean	median	min	max
RMS [mm]	2.22	1.94	0.64	5.00	1.03	0.98	0.69	1.64
RotE [$^\circ$]	2.36	2.23	0.37	6.16	1.10	1.03	0.11	2.09
Time [s]	28.85	28.05	0.60	>60	0.38	0.35	0.05	0.95

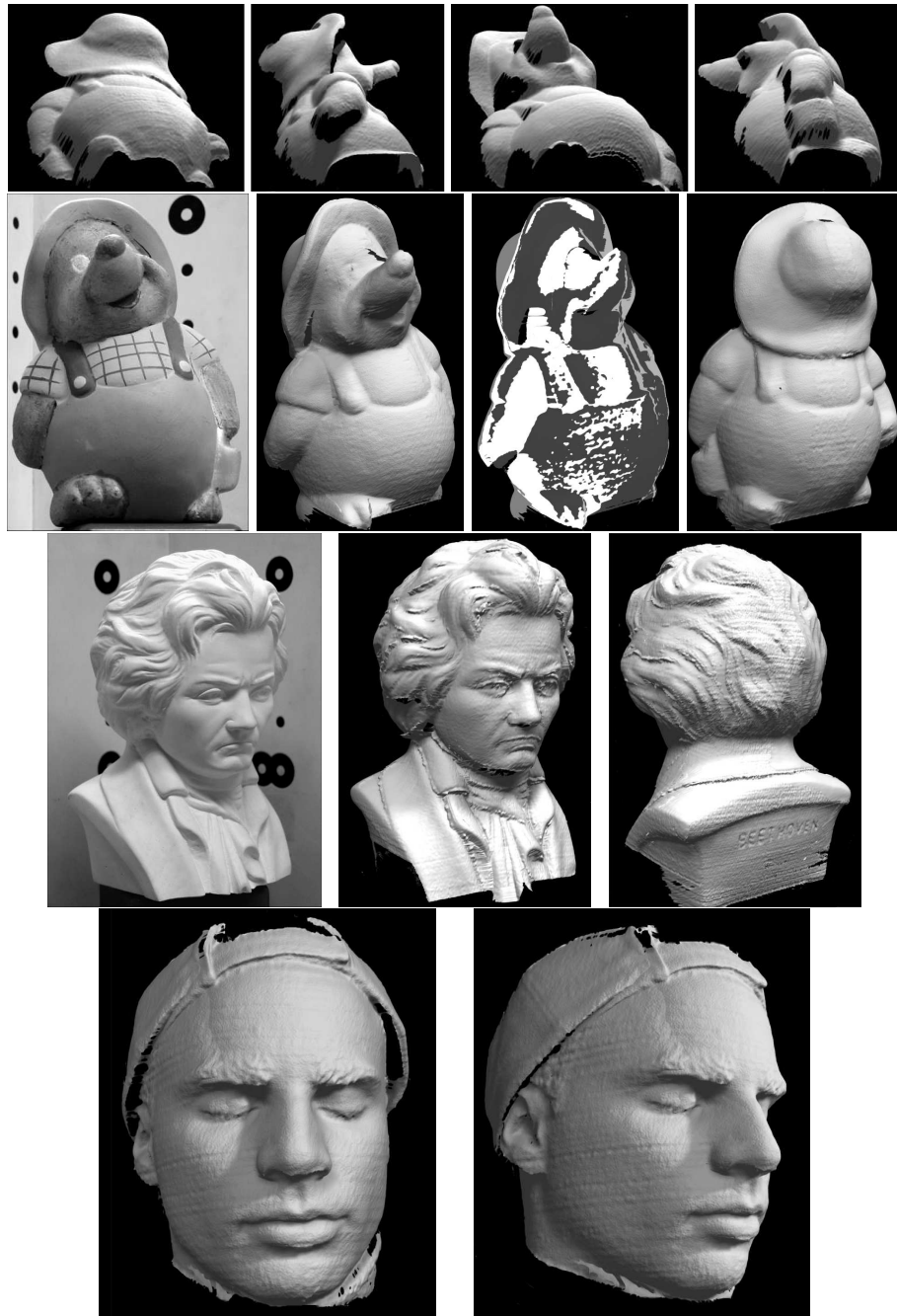


Fig. 5. (Row 1) scans of a test object. (Row 2) camera image; registered scans; each scan in a different color; back view. (Row 3) camera image and two views consisting of four registered scans. (Row 4) two views of a head consisting of two registered scans.

penetration and the fusion edges. The third row shows a Beethoven bust (height 170 mm) – a camera image, and the registration result from two different sides. Although it is easily possible to texturize the scenes using the camera images, we only show the untexturized scans to demonstrate that even small shape details are acquired accurately. For example, the “BEETHOVEN” engraving on the back can be read very well although it is only 0.4 to 0.7 mm in depth. In the last row we present scan and registration results of one of the authors’ head. He was sitting on a chair in the corner of a room, the walls served as background for laser calibration. Between the two scans, the camera has not been moved, instead the person has turned his head.

The horizontal artifacts (stripes) that are visible on flat surfaces in most scan results are caused by slight inaccuracies during on-line laser calibration. They can be effectively reduced using space and/or time filtering. Gaps in the scanned surface appear at places the camera cannot see (“shadow”), when the corresponding area is too dark, or when it has been scanned so quickly that the laser did not intersect it in any image.

With these impressive results, we have demonstrated that simple low-cost equipment is sufficient to build up a system for 360°-object-reconstruction, which is superior to other techniques (e.g. scanning flexibility and registration efficiency).

References

1. Pipitone, F.J., Marshall, T.G.: A wide-field scanning triangulation rangefinder for machine vision. *International Journal of Robotics Research* **2**(1) (1983) 39–49
2. Hall, E.L., Tio, J.B.K., MCPerson, C.A.: Measuring curved surfaces for robot vision. *Computer* **15**(12) (1982) 42–54
3. Blais, F.: Review of 20 years range sensor development. *Journal of Electronic Imaging* **13**(1) (2004)
4. Zagorchev, L., Goshtasby, A.: A paintbrush laser range scanner. *Computer Vision and Image Understanding* **101** (2006) 65–85
5. Besl, P.J., McKay, N.D.: A method for registration of 3-D shapes. *IEEE Trans. Pattern Anal. Machine Intell.* **14**(2) (1992) 239–258
6. Krebs, B., Sieverding, P., Korn, B.: A fuzzy icp algorithm for 3d free form object recognition. In: *International Conf. on Pattern Recognition.* (1996) 539–543
7. Dalley, G., Flynn, P.: Pair-wise range image registration: a study in outlier classification. *Comput. Vis. Image Underst.* **87**(1-3) (2002) 104–115
8. Chua, C.S., Jarvis, R.: Point signatures: A new representation for 3d object recognition. *International Journal of Computer Vision* **25**(1) (1997) 63–85
9. Papaioannou, G., Theoharis, T.: Fast fragment assemblage using boundary line and surface matching. In: *IEEE/CVPR Workshop on Applicat. of Computer Vision in Archaeology.* (1999)
10. Krebs, B., Korn, B., Wahl, F.M.: Plausibilistic preprocessing of sparse range images. In: *Proc. of the 8th Int. Conf. on Image Anal. and Processing.* (1995) 361–366
11. Johnson, A., Hebert, M.: Recognizing objects by matching oriented points. In: *Proc. IEEE Conf. Computer Vision and Pattern Recognition.* (1997) 684–689
12. Schön, N., Häusler, G.: Automatic coarse registration of 3d surfaces. In: *Vision, Modeling, and Visualization 2005.* (2005)

13. Barequet, G., Sharir, M.: Partial surface matching by using directed footprints. In: 12th annual symposium on Computational geometry. (1996) 409–410
14. Silva, L., Bellon, O.R.P., Boyer, K.L.: Robust Range Image Registration Using Genetic Algorithms and the Surface Interpenetration Measure. Volume 60 of Machine Perception Artificial Intelligence. World Scientific (2005)
15. Winkelbach, S., Rilk, M., Schönfelder, C., Wahl, F.M.: Fast random sample matching of 3d fragments. In: Pattern Recognition, 26th DAGM Symposium. Volume 3175 of Lecture Notes in Computer Science., Springer (2004) 129–136
16. Fischler, M.A., Bolles, R.C.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Communications of the ACM **24**(6) (1981) 381–395
17. Tsai, R.Y.: An efficient and accurate camera calibration technique for 3d machine vision. In: IEEE Conf. Computer Vision and Pattern Recognition. (1986) 364–374
18. Wahl, E., Hillenbrand, U., Hirzinger, G.: Surflet-pair-relation histograms: A statistical 3d-shape representation for rapid classification. In: Proc. 4th International Conf. on 3-D Digital Imaging and Modeling (3DIM'03). (2003) 474–481
19. Weisstein, E.W.: Birthday Attack. (From MathWorld—A Wolfram Web Resource. <http://mathworld.wolfram.com/BirthdayAttack.html>)
20. Friedman, J.H., Bentley, J.L., Finkel, R.A.: An algorithm for finding best matches in logarithmic expected time. ACM Trans. on Math. Software **3**(3) (1977) 209–226