Accurate and Robust Registration for In-hand Modeling

Thibaut Weise[1] Bastian Leibe[1] Luc Van Gool[1,2]
[1]ETH Zurich
Zurich, Switzerland
[2]KU Leuven
Leuven, Belgium
{weise,leibe}@vision.ee.ethz.ch vangool@esat.kuleuven.be

Abstract

We present fast 3D surface registration methods for in-hand modeling. This allows users to scan complete objects swiftly by simply turning them around in front of the scanner. The paper makes two main contributions. First, we propose an efficient method for detecting registration failures, which is a vital property of any automatic modeling system. Our method is based on two different consistency tests, one based on geometry and one based on texture. Second, we extend ICP by three additional fast registration methods for both coarse and fine alignment based on both texture and geometry. Each of those methods brings in additional information that can compensate for ambiguities in the other cues. Together, they allow for the robust reconstruction of a large variety of objects with different geometric and photometric properties. Finally, we show how both failure detection and fast registration can be combined in a practical and robust in-hand modeling system that operates at interactive frame rates.

1. Introduction

3D scanning and modeling of real world objects is attractive for a variety of applications, including entertainment, cultural heritage and architecture. Most scanning systems capture 3D surface patches from different viewpoints *e.g.* using a turntable, and the different parts are stitched together in an offline process to form a coherent mesh. However, when scans are only registered offline, such a process risks leaving holes in the reconstruction in places where the scanner could not reach from the predetermined viewpoints. Moreover, the offline processing means that those holes may only be detected much later. Filling in the missing data then requires another scanning session, which is a tedious process and which may in some cases even be impossible, since the object is no longer accessible.

In-hand modeling is an appealing concept, since it circumvents this problem. By allowing the user to interactively see and control the object, while it is being reconstructed online, he can directly fill in any remaining holes [22, 13, 26]. Previous systems have shown that in-hand scanning is feasible in principle. However, in order to be useful in practical applications, it needs to be robust to a large variety of objects with different geometric and photometric properties.

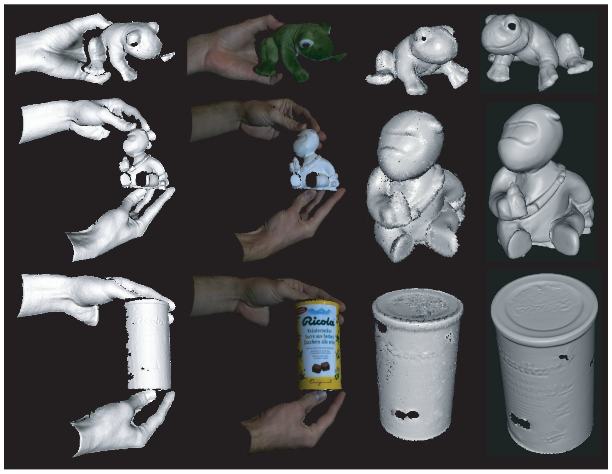


Figure 1. In-hand modeling of different objects: Typical scan, textured scan, online preview model and final post-processed model for *frog*, *boris* and *ricola*.

Especially symmetrical and highly textured objects pose problems to current in-hand scanning systems, which only use geometry-based ICP for registration [22, 13, 26]. Moreover, inevitable registration failures on such objects are not detected by those systems, leading to reconstruction artifacts in the final model [25].

In this paper, we bridge the gap between feasibility and practicality by the following contributions. 1) We propose an efficient method for detecting registration failures based on two different consistency tests using texture and geometry information. 2) We propose three additional coarse and fine registration methods based on texture and geometry to complement ICP. Together, these methods cover a wide variety of object properties. Fine texture registration is included into ICP using point-to-plane constraints along the gradient direction of texture edges. Coarse texture registration is performed using texture features for RANSAC-like transformation estimation and testing. For coarse geometric registration, we describe a novel method based on snapshots and spin-images combined with hypothesis verification using geometric consistency. 3) As speed is of prime importance for interactive modeling, we show how the proposed methods can be optimized for real-time performance on the GPU. 4) We demonstrate how the different components can be combined into a practical in-hand scanning system and report results on a variety of objects with different geometric and photometric properties.

The paper is structured as follows. Section 2 discusses related work. After that, Section 3 introduces our novel failure detection approach based on geometric and texture consistency. Section 4 describes our proposed registration methods, and Section 5 evaluates them in detail. Finally, Section 6 shows how all components are integrated into a practical in-hand scanning system.

2. Related Work

A comprehensive overview on methods required for 3D modeling is given by [3]. Recently, interactive in-hand 3D modeling systems have been demonstrated by [22, 13, 26].

Registration of the individual surface patches is a crucial part of the modeling pipeline, and ICP [4, 7] and its variants [23, 10] are the best-known technique for registration, assuming that an approximate alignment is known. An extension to include texture has been presented by [16]. Registration based on texture correlation additionally to ICP has been proposed by [2].

When no approximate alignment is known, coarse registration is required. Using texture features for coarse registration has been proposed by [24]. There is a wealth of geometry-based methods, and [6] give a comprehensive overview. Recent methods include [9, 17, 19]. Geometric consistency for coarse registration has been used by [5].

Registration failure detection is necessary for any automated system. [12] proposed a geometric consistency measure for failure detection within a multiview registration framework. To our knowledge texture consistency has not yet been used for verification.

Scanning Setup. For shape and texture capture, we use a phase-shift structured-light scanner as in [28, 27]. It uses a modified DLP projector for projecting three sinusoidal patterns into the working volume and a synchronized camera for recording the resulting intensity images. A dense reconstruction is obtained at 20-30 fps through triangulation of the measured phase values for each pixel with the known projection. For this, the wrapped phase is calculated directly from the intensities, and phase unwrapping determines the phase period. A texture image is calculated as the average intensity of the three patterns.

3. Registration Consistency

Automatic modeling systems need some form of registration failure detection, as already a few misregistered scans may lead to distorted models [25]. Similarly, interactive modeling [23, 13, 26] relies on correct sequential registration. Yet, registration often fails in practice without being detected. This is because many objects of practical interest exhibit symmetries which may lead to misregistrations with low error.

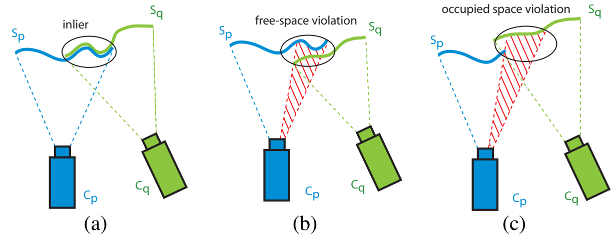


Figure 2. Geometric consistency: a) Consistent surfaces as S_p and S_q are close in C_p. b) Free space violation as S_q occludes S_p in C_p. c) Occupied space violation as S_q is not seen in C_p.

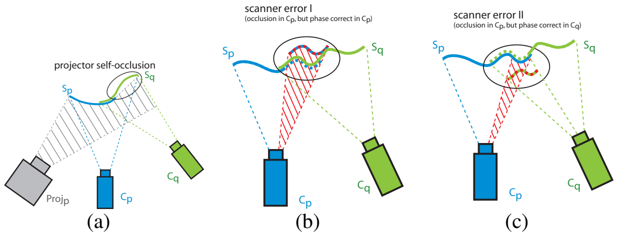

projector self-occlusion
scanner error I
(occlusion in Cp, but phase correct in Cp)
scanner error II
(occlusion in Cp, but phase correct in Cq)
Sp
Sq
Cp
Cq
Projp
(a)
(b)
(c)

Figure 3. a) OSV due to projector occlusions are handled by projection. a) *type-I scanner error*: S_q occludes (incorrect) S_p, but phase is consistent in C_p. b) *type-II scanner error*: (incorrect) S_q occludes S_p, but phase is consistent in C_q.

In the following, we propose an accurate and robust registration consistency measure that describes the quality of the registration and that can be used for failure detection. The consistency measure is based on both geometry and texture, thus integrating all available information.

3.1. Geometric Consistency.

Geometric consistency of a registration is measured as the consistency of two surfaces along the cameras' lines of sight. Here, we extend the method presented by [12]. Consider for example the two registered surface scans S_p and S_q in Fig. 2 that have been captured by cameras C_p and C_q respectively. The registration is consistent (IN) for camera C_p if the overlapping parts of S_p and S_q are at similar range (Fig. 2(a)). A free-space violation (FSV) occurs in C_p when the registered surface S_q occludes the observed surface S_p (Fig. 2(b)). An occupied space violation (OSV) occurs when surface S_q is falsely not observed in C_p (Fig. 2(c)). If S_p occludes S_q in C_p, it is treated as *don't care* (GND) as it does not violate the observation in C_p.

The geometric consistency ratio is calculated by rendering the registered surface S_q into camera C_p using z-buffering and by then comparing the rendered range image (S_q) pixel-by-pixel to the captured range image (S_p). Each pixel x in C_p is classified by comparing the two depth val-

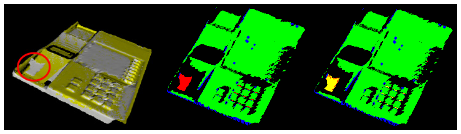


Figure 4. a) Two registered scans. Geometric consistency b) without and c) with phase consistency compensation.

ues $z_p(x)$ (from S_p) and $z_q(x)$ (from S_q):

$$x \in \begin{cases} X_{IN}(p,q) & \text{if } |d_{(p,q)}(x)| < t_{IN} \\ X_{FSV}(p,q) & \text{if } d_{(p,q)}(x) < -t_{IN} \\ X_{OSV}(p,q) & \text{if } \exists z_q(x) \ \wedge \ \neg\exists z_p(x) \\ X_{GND}(p,q) & \text{if } d_{(p,q)}(x) > t_{IN} \ \vee \ \neg\exists z_q(x) \end{cases} \tag{1}$$

where $d_{(p,q)}(x) = z_q(x) - z_p(x)$ and t_{IN} is an inlier threshold. We record the FSV and OSV ratios as follows:

$$R^{pq}_{FSV} = \frac{|X_{FSV}(p,q)| + |X_{FSV}(q,p)|}{|X_{IN}(p,q)| + |X_{IN}(q,p)|} \tag{2}$$

$$R^{pq}_{OSV} = \frac{|X_{OSV}(p,q)| + |X_{OSV}(q,p)|}{|X_{IN}(p,q)| + |X_{IN}(q,p)|} \tag{3}$$

We classify a registration T_{pq} as correct whenever both ratios are below predefined thresholds:

$$T_{pq} \text{ is} \begin{cases} \mathit{valid} & \text{if } R^{pq}_{FSV} < t_{FSV} \ \wedge \ R^{pq}_{OSV} < t_{OSV} \\ \mathit{invalid} & \text{otherwise} \end{cases} \tag{4}$$

Fig. 4(b) shows an example classification where inliers are marked in green, X_{FSV} are marked in red, X_{OSV} are marked in blue, and X_{GND} are kept black.

This framework has been used successfully in several applications [12, 20]. However, it does not yet take into account how the range image was created. In the following, we improve on it by using additional information available from phase-shift range scanners.

Invalid Free Space Violation. FSV is a primary sensor space violation, and the ratio threshold should be set as restrictive as possible. Whenever the reconstructed surfaces are error-free, a correct registration will create no FSV. However, depending on the range sensor, the reconstructed surfaces might contain outliers and thus create *invalid* FSVs for an otherwise correct registration.

For our scanning setup, the phase unwrapping may assign the wrong period to a pixel, and may thus record too large or too small a depth value. When the depth is too large, a *type-I scanner error* may occur (Fig. 3(b)): the correct surface S_q occludes the wrong surface S_p and generates an FSV. However, the estimated[1] wrapped phase of S_q will be consistent with the wrapped phase in C_p, and thus the FSV can be invalidated. When the depth is too small, a *type-II scanner error* may occur (Fig. 3(c)): the incorrect surface S_q occludes the true surface S_p. However, S_p and S_q are phase-consistent in C_q, thus invalidating the FSV.

Hence, for every FSV pixel, the phase consistency is checked in both C_p and C_q, and the FSV is invalidated if any of the two is phase-consistent. Fig. 4(a) shows an example registration of two surfaces. The encircled gray surface patch is a *type-I scanner error*: the assigned depth is too small. Hence, it creates FSV pixels, as shown in red in Fig. 4(b). Phase consistency is used to detect and remove *invalid* FSVs, as shown in yellow in Fig. 4(c). This implies that we can recover from errors made by the scanner.

Invalid Occupied Space Violation. For our system the following reasons for *invalid* OSVs can be identified (the first four yield no reconstruction):
1) The surface S_q is occluded in the projector (Fig. 3(a)).
2) The surface is too oblique to either camera or projector.
3) Saturation or too dark surface cause problems.
4) The scanner assigns no period to a pixel in C_p (similar to *type-I scanner error*).
5) The scanner performs a wrong reconstruction in S_q by assigning the wrong period (*type-II scanner error*).

We can correct for most of the reasons above, thus making the remaining OSV a valid consistency measure. Projector occlusions are removed by projecting both S_p and S_q into the projector $Proj_p$. Each OSV pixel is thus checked whether it is occluded by $Proj_p$. Oblique OSV pixels are removed by checking the surface normal in S_q. OSVs due to saturation are removed by checking intensity at the pixel in C_p. OSV pixels due to scanner errors I and II are handled similar to *invalid* FSV.

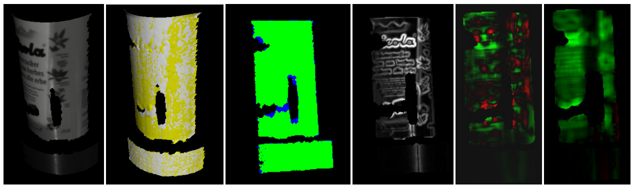
OSV due to low signal strength are not corrected for. Moreover, if there are overlapping scanner errors in both scans, these go undetected. However, registration failure should be a conservative measure in any automatic registration system. Thus, too many *invalid* OSV (or *invalid* FSV) will at most result in a failure detection for an otherwise correct registration, which is acceptable in many situations. In practice, the OSV ratio can be made less restrictive than the FSV ratio. False OSV and FSV invalidations are possible due to random phase consistency, but are negligible in practice.

3.2. Texture Consistency.

For symmetric objects such as a cylinder, surface geometry alone cannot yield an unambiguous registration. Thus, no geometry based consistency measure can detect registration failure in all cases (see Fig. 5). In contrast, texture can not only help for registration (see Sec. 4.2), but also for measuring the consistency of the registration.

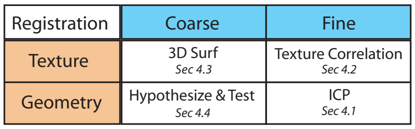
For each intensity image I_p and I_q, we calculate the gradient magnitude using simple Gaussian derivative kernels (Fig. 5(d)). Similar to geometric consistency, the texture of surface S_q is compared against the texture of S_p in C_p's image space. For each pixel $x \in C_p$, the gradient magnitudes $g_p(x)$ and (projected) $g_q(x)$ are compared with local nor-

[1] We can estimate the phase by projecting the surface point into the projector and assigning the corresponding phase.



(a) (b) (c) (d) (e) (f)

Figure 5. a) Scan of a cylinder. b) Bad registration despite c) good geometric consistency. d) Magnitude image. e) Bad registration with low texture consistency (neg. evidence in red, pos. in green). f) Correct registration with high texture consistency.



Registration	Coarse	Fine
Texture	3D Surf *Sec 4.3*	Texture Correlation *Sec 4.2*
Geometry	Hypothesize & Test *Sec 4.4*	ICP *Sec 4.1*

Figure 6. Registration methods are divided into whether a coarse or fine alignment is required and whether texture or geometry are used as cues.

malized cross-correlation (NCC). NCC is weighted by the local maximum magnitude, thus emphasizing textured areas. Unweighted NCC ranges from -1 to 1; thus negative values give negative evidence, and positive values give positive evidence for the registration (Fig. 5(e)). Both are aggregated for the whole image, and the ratio negative to positive evidence R_{TEX} is used as texture inconsistency measure.

We exclude pixels close to depth and normal discontinuities based on the variance of the depth derivative. Specularities on the object reduce texture consistency, but can be detected given the known scanner geometry and surface normals. Hence, we include a term to reduce the correlation weight at surface points with high specular likelihood.

4. Registration Methods

Registration is a crucial part of any modeling system, bringing all surface patches into a common coordinate system. Two types of registration methods can be distinguished for in-hand scanning: fine registration for finding exact alignments between successive scans and coarse registration for reinitializing the scanning procedure after the object has been lost or moved out of the scanner. Both types can be based on geometry or texture (see Fig. 6). Current in-hand scanning systems only use geometry-based ICP [23, 13, 26].

However, only a combination of different registration types can lead to robust and accurate registration for a wide variety of objects. In addition, the methods should be fast enough for interactive registration tasks in an in-hand scanning system, or for an automatic multi-view registration system. In the following we propose three additional methods for fast, robust, and accurate registration.

4.1. Fine Geometry

ICP [4, 7] is the best-known technique for pairwise surface registration. ICP performs fine geometric registration assuming that a coarse registration transformation T_{pq}^0 is already known. Many different variants of ICP have been proposed; a comprehensive overview can be found in [23]. For real-time registration we use the method as described by [13]. Fast ICP uses projection along the line of sight of the range sensor for closest-point lookup, thus reducing the closest-point search from $O(\log n)$ to $O(1)$. Instead of minimizing the point-to-point distance, it uses the point-to-plane error metric as introduced by [7]:

$$\sum n(q_j)\left(T_{pq}^{k+1} p_i - q_j\right) \tag{5}$$

where $n(q_j)$ is the estimated normal at q_j, and T_{pq}^{k+1} is the transformation minimizing this distance after k iterations for point sets $P{=}\{p_1, .., p_N\}$ and $Q{=}\{q_1, .., q_M\}$.

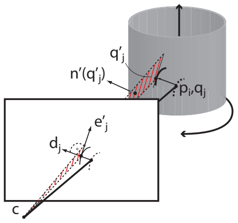


Figure 7. Fine texture registration is used for accurate registration of geometrically underconstrained alignments, such as for a cylinder. Closest point search is performed along the gradient direction d_j, and the plane $\Pi = \{c, n(q_j^t)\}$ is added into ICP.

4.2. Fine Texture

As pure ICP uses no texture information, geometric symmetries result in ambiguities. For example, a cylindrical object has two unconstrained degrees of freedom: translation along and rotation around the principal axis (Fig. 7). Texture can often break the symmetry. Moreover, whereas ICP is particularly suited for optimizing along the range sensor's line of sight, texture is particularly suited for constraining the optimization perpendicular to the line of sight, thus nicely adding complementary information also for asymmetric surfaces.

ICP with texture has been used before by modifying the closest-point search to include texture [16]. However, this is not compatible with fast projective lookup. On the other hand, assuming we already have a good registration, we can project and search the local neighborhood for a compatible point [2]. Using the compatible point directly with the point-to-point distance metric would provide enough constraints. However, for many objects, texture edges are predominant, and finding the compatible point is again underconstrained, similar to the aperture problem of optical flow.

We take advantage of the aperture ambiguity by searching for the most compatible point q'_j only along the gradient direction d_j (see Fig. 7). We use the best point as point-to-plane constraint, which can be conveniently added into the ICP error metric. Instead of using the surface normal of the most compatible point, we use the normal $n'(q'_j)$ of the plane formed by the sensor center c and e'_j, which is the line

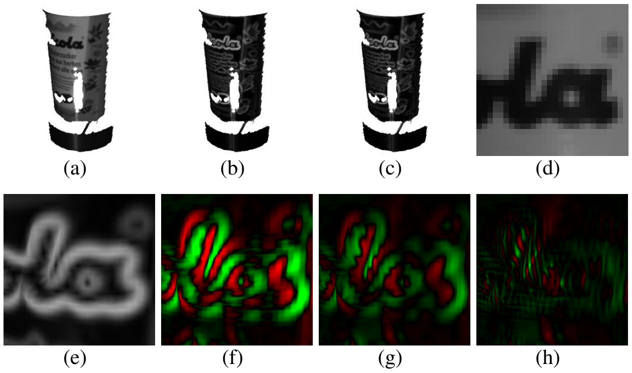

(a) (b) (c) (d)
(e) (f) (g) (h)

Figure 8. a) Texture of 1st scan. b+c) Gradient magnitude of 1st and 2nd scans projected onto 1st scan. d) Close-up texture image e) Corresponding magnitude image f,g,h) Magnitude difference image for ICP, coarse texture + ICP, and fine texture registration. The improvement can be clearly perceived as a reduction of the differences (red positive, green negative difference).

perpendicular to the gradient direction (on the image plane). This leads to the following ICP error metric:

$$\sum n(q_j)\left(T_{pq}^{k+1}p_i - q_j\right) + \lambda\sum w'_j n'(q'_j)\left(T_{pq}^{k+1}p_i - q'_j\right) \quad (6)$$

where w'_j is a compatibility weighting factor and λ is used to control the influence of texture within ICP.

We use a 3D object space error metric instead of the correct 2D image space error. The inaccuracy is negligible, as fine registration assumes an initialization close to the correct registration. Thus, the surface distance to the camera will remain approximately constant, and the object space error will be the scaled image space error.

Compatible Point Search. For each p_i, we search for the most compatible point q'_j using correlation with a 5×5 window along d_j. Similar to the texture consistency test (Sec. 3.2), we correlate the gradient magnitudes to be more robust to intensity variations. We do not need the concept of *negative* evidence here and can therefore use SSD instead of NCC for efficiency. Subpixel accuracy is achieved using quadratic interpolation of the SSD. For each compatible point q'_j, the correlation weight w'_j is calculated based on the correlation function and the gradient magnitude. Similar to texture consistency, the influence of parts with high specular likelihood and/or geometric variance is reduced by reducing the correlation weight w'_j. Texture distortions due to surface transformations are implicitly handled by projecting each point in the correlation window independently.

4.3. Coarse Texture

Texture features can also be used for coarse registration and thus as initialization for ICP. We use the following basic approach: 1) extract interest points, 2) find correspondences, 3) generate transformation hypotheses, 4) test hypotheses and keep the one with most inliers.

We detect interest points using Harris corners [11] on the texture image and calculate a SURF descriptor [1] for each feature point. Scale invariance is inherently obtained by scaling the descriptor window size with the depth of the feature point; orientation invariance is achieved by determining the dominant orientations as in SIFT [18]. We do not warp the texture based on 3D geometry as described in [24], as this did not improve registration in our experiments.

For two scans P and Q, we get a set of feature points $F_P = \{f_i^p, i = 1..N\}$ and $F_Q = \{f_j^q, j = 1..M\}$. The 3D position v_i^p of f_i^p is obtained from the range images

$$v_i^p = d(f_i^p)K_C^{-1}x(f_i^p) \quad (7)$$

where K_C are the camera intrinsics, $x(f_i^p)$ is the homogeneous feature coordinate, and $d(f_i^p)$ is its depth.

Feature matching based on Euclidean distance is used to create a set of correspondences $K_{ij} = (f_i^p, f_j^q)$. In order to estimate a transformation, we need three valid feature correspondences. In theory, we could also use the estimated normal and feature orientation to reduce the necessary correspondences to one. However, both orientation and estimated normal are often noisy, particularly as texture corners frequently coincide with geometric corners.

Usually, RANSAC [8] is used to remove outliers. RANSAC generates a random set of transformation hypotheses and chooses the transformation T_{pq} with most inliers. All inliers are then used to calculate the optimal alignment. A feature correspondence K_{ij} is counted as inlier iff $||T_{pq}v_i^p - v_j^q|| < t_D$, where t_D is a user-defined threshold.

Similar to [9], we use the implicit distance RMS error ($dRMS$) to speed up the transformation calculation. Then, for two inlier feature correspondences K_{ij} and K_{kl}:

$$dRMS(K_{ij}, K_{kl}) = \left(||v_i^p - v_k^p|| - ||v_j^q - v_l^q||\right)^2 < 2t_D \quad (8)$$

The smaller the $dRMS$, the more distance-compatible are two correspondences. Thus, we speed up the *valid* hypothesis generation time by only using distance-compatible feature correspondences. In addition, we employ the heuristic that an inlier correspondence K_{ij} will most likely have another inlier correspondence K_{kl} as most compatible counterpart. We adopt an iterative greedy strategy. First we find for a feature correspondence the *two* most compatible counterparts. We then get the most distance-compatible triple minimizing the combined $dRMS$ error (Fig. 9). Thus, we generate one hypothesis per feature correspondence and keep the transformation T_{pq} with most inliers. T_{pq} can then be used as initialization of ICP. In addition, inlier features of T_{pq} can be integrated into the ICP framework.

All of the above steps, namely corner detection, SURF descriptor calculation, feature matching, transformation hypothesis generation, and inlier test are implemented on the latest GPU (NVidia Geforce 8800 GTX) using both CG and CUDA. Thus, coarse registration based on texture features takes about 10 ms for two scans of about 30000 vertices, image size 640×480, 500 features, and 250 correspondences.

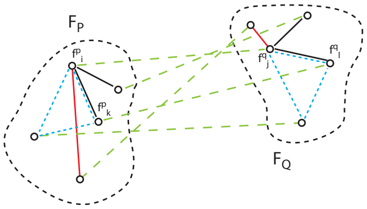


Figure 9. Correspondences of two features sets F_P and F_Q are shown in green. The *two* most distance-compatible correspondences of $K_{ij} = (f_i^p, f_j^q)$ are shown in black, an incompatible correspondence red, and the most compatible triple in blue.

4.4. Coarse Geometry

In case no texture features or approximate alignment are available, coarse registration based on geometry needs to be performed. Previous methods we are aware of are too slow for interactive applications or lack robustness. In the following, we therefore propose a new algorithm that is particularly fast (200-300ms). This algorithm adopts a hypothesize & test framework for geometric registration, where most steps are efficiently implemented on the GPU.

For hypothesis generation we combine the idea of snapshots [19] with spin-images [14]. We uniformly sample the surface points of a scan and create a snapshot for each sample. A snapshot is a synthetic rendering of only the local surface, as seen from the perspective of a virtual camera that is placed at a certain distance along the sample normal [19]. This way, self-occlusions of the surface are automatically handled. Instead of using the snapshots directly as descriptors as in [19], we create a rotation invariant representation similar to spin-images. Our modified descriptor uses the pixel-radius and depth as given by the snapshots.

One descriptor is created for each sample. For two scans, the feature descriptors are matched, and transformation hypotheses are generated from the best distance-compatible pairs (similar to texture registration). In this case, we only use two features and their normals, as many samples will lie on smooth parts and the normal will often be correct.

Uniformly sampling the surface points implies that we may not sample the same points on both scans. In practice, when enough samples are used (here 2500), neighboring points often have similar descriptors, and matching will pick a point close to the correct match. The resulting small error is negligible for coarse registration, and two correct correspondences will still likely be distance-compatible.

Instead of using inlier counting, we test each transformation hypothesis for geometric consistency (see Sec. 3.1) on the actual scan data. The advantage is that this test is independent of the actual number of inliers. Furthermore, only hypotheses that are consistent with the scans are chosen. We combine the FSV and OSV ratios (Eq. 2 & Eq. 3) to get an inconsistency score for each hypothesis h:

$$R^{ij}(h) = \mu R^{ij}_{FSV} + R^{ij}_{OSV} \tag{9}$$

where μ determines the relative weight of R^{ij}_{FSV} and R^{ij}_{OSV}. We keep the hypothesis with the lowest score. The inlier threshold t_{IN} of Eq. 1 is increased, since larger registration errors will be tolerable for coarse registration. For efficiency reasons, we do not check for *invalid* FSV or OSV. A further speedup is obtained by performing the consistency checks on a subsampled version of the scans. Note that coarse geometry is sufficient, as small geometric detail will have a small effect on the consistency check.

Object	ICP	Coarse Geometry	Coarse Texture
boris	36 (0 / 4.2)	**74** (**0** / 3.0)	42 (0 / 5.3)
frog	56 (0 / 2.6)	**66** (**0** / 2.9)	38 (0 / 5.0)
coati	58 (0.2 / 0.1)	**74** (**0** / 0.7)	46 (0 / 0.9)
marvin	50 (0 / 1.4)	**56-70** (**0** / 0.6)	44 (0 / 3.7)
pot	10 (16.1 / 0)	14 (21.8 / 0)	**82** (12.9 / **0**)
ricola	4 (100 / 1.2)	4 (98 / 2.0)	**60** (72.2 / **2.8**)

Table 1. Registration results for the objects from Fig. 10(a). The first number indicates the rotation angle at which the registration method fails with 50% probability. The values in brackets show the false positive rates for geometry and texture consistency tests.

Both hypothesis generation and hypothesis testing are independent, and each part can be used within other coarse registration frameworks.

5. Registration Results

Texture registration significantly improves both robustness and accuracy. Fig. 8 shows the gradient magnitude difference images of two scans which have been registered using ICP, coarse texture + ICP, and fine texture enhanced ICP. The results are considerably better for ICP + coarse texture than for ICP and best for fine texture registration.

For a quantitative evaluation, we present registration tests on a variety of objects with different geometric and photometric properties (shown in Fig. 10(a)). Controlled input data was generated by rotating or translating each object on a turntable. Ground truth alignments were generated using sequential registration with multi-view optimization. In order to generate ground truth for the rotationally symmetric *'ricola'* can, a marker was placed on its top for scanning, but was later removed from the scans for testing.

Rotation. Tab. 1 compares the different methods' results when the test objects are rotated. Each scan was registered with all other scans, and the difference to the ground truth was recorded. The results show how far the object can be rotated until more than 50% of the registrations fail, where failure is defined as a deviation of more than $1°/1\text{mm}$ in rotation/translation from the ground truth ($3°/3\text{mm}$ for *'ricola'* and *'pot'*). The results show that both coarse texture and geometric registration improve on pure ICP and that the most appropriate method depends on the object's properties.

Tab. 1 also displays the false positive rates of both geometric and texture consistency. This rate is particularly important, as it signifies how many failed registrations are classified as correct. All objects have below 0.2% false

Method	ms
ICP (10K samples, 20 (5) iter)	17.5 (6.5)
Texture Correlation	12
3D SURF (500 features, 250 matches, 200 inliers)	10
Hyp. & Test (2304 samples, 1K hypotheses)	220
Geometric Consistency	9
Texture Consistency	7.8

Table 2. Timings for registering two pot scans, with each scan consisting of $\sim$30K vertices (coarse: 3500).

positive rate, except for 'ricola' at 2.8%, which can be explained by the specularity of the object. (False negative rates were between 5-10% for this experiment).

Translation. When comparing performance under translation, we found that ICP fails for all objects at $\geq$ 10-40mm translation. Coarse registration using texture or geometry is not affected, as it is implicitly position independent.

Examples. Fig. 10 visualizes these results for two typical examples. Fig. 10(b) shows the large performance increase due to coarse texture registration on the rotationally symmetric *'pot'*. In this example, coarse texture registration can successfully align two surfaces that are rotated by up to $80°$, whereas ICP fails already at about $5°$. Fig. 10(e) displays the texture ratio R_{TEX} for the corresponding registration methods. Whenever registration fails, R_{TEX} increases beyond the dashed line and detects the failure reliably.

The improvements due to coarse geometry registration are shown in Fig. 10(c) for *'boris'*: the funnel of convergence for rotation is increased from about $30°$ (ICP) to up to $70°$. Fig. 10(f) demonstrates that the FSV ratio R_{FSV} can reliably detect the registration failures.

Finally, Fig. 10(d) verifies that transformation hypothesis generation using the most distance-compatible triangles is as good as standard RANSAC (50K hypotheses). In addition, registration only fails when the number of inliers goes below 10 (out of $\sim$200 correspondences and 500 features).

Timings for all methods and consistency tests can be found in Tab. 2. All methods require typically around 10 ms, except for coarse registration which takes between 200 to 300 ms. ICP takes 17.5 ms for 20 iterations, but in practice only about 5 iterations are required.

6. In-hand Modeling

Our in-hand modeling system is akin to the approach introduced by [22]. This system however only uses ICP and requires users to manually align the reconstructed object to displayed scans for reinitialization. Moreover, this system cannot handle highly textured and/or symmetric objects. In contrast, we propose to employ all four registration methods described in Sec. 4 and combine them with automatic failure detection.

First, coarse texture registration is performed at each step and is used whenever enough inliers are available. ICP incorporating the coarse texture features is executed next. Finally, fine texture registration is used for refining the alignment. Coarse geometry registration is only performed when no features are available and the previous registration failed. Coarse geometry turns out to be particularly useful for reinitialization, *e.g.* when the user takes out the object, investigates the preview model, and then continues to scan. It is only applied about once a second to keep the system responsive (as it takes up to 300 ms for coarse registration).

Geometric and texture consistency are used for failure detection, thus removing registration errors that would make the modeling system "useless" due to sequential error propagation. The system is adaptive in the sense that it performs registration based on texture whenever enough features are available. Fine texture registration and the texture consistency test are currently enabled manually, but for future work this could be replaced by an adaptive scheme.

As the user's hands will be visible during in-hand scanning, we use skin color detection based on [15] to efficiently remove the hands from the geometry. For each depth sample, the color is used in conjunction with a lookup table to decide whether it is a skin pixel. Median filtering is used for robustness. Finally, foreground-background segmentation [21] on the phase images removes the necessity of having a black background.

Altogether, the in-hand modeling system runs at about 10 fps for an average scan size of 30,000 triangles. About one third of the computation time is spent on data capture. Some typical modeling results can be found in Fig. 1. Note that the final model is integrated using offline processing [22]. Videos showing the system in action are at `http://www.vision.ee.ethz.ch/~weiset/cvpr08`.

7. Conclusion

We have presented a set of fast registration methods using both geometric and photometric cues. For robust and accurate registration, all available cues need to be integrated, and we proposed three additional registration methods to complement ICP for textured and symmetric objects. In addition, we have proposed two efficient consistency tests for registration failure detection, which is a crucial capability for automated model integration. As we have shown, all of those methods can be efficiently integrated, creating a truly practical in-hand modeling system.

Acknowledgments. This research has been funded in parts by EU projects ImmerSence (IST-2006-027141) and CHIRON (MEST-CT-2004-514539).

References

[1] H. Bay, T. Tuytelaars, and L. Van Gool. SURF: Speeded up robust features. In *ECCV (1)*, 2006.

[2] F. Bernardini, I. M. Martin, and H. E. Rushmeier. High-quality texture reconstruction from multiple scans. *IEEE Trans. Vis. Comput. Graph*, 7:318–332, 2001.

[3] F. Bernardini and H. Rushmeier. The 3D model acquisition pipeline. *Comp. Graph. Forum*, 21:149–172, 2002.

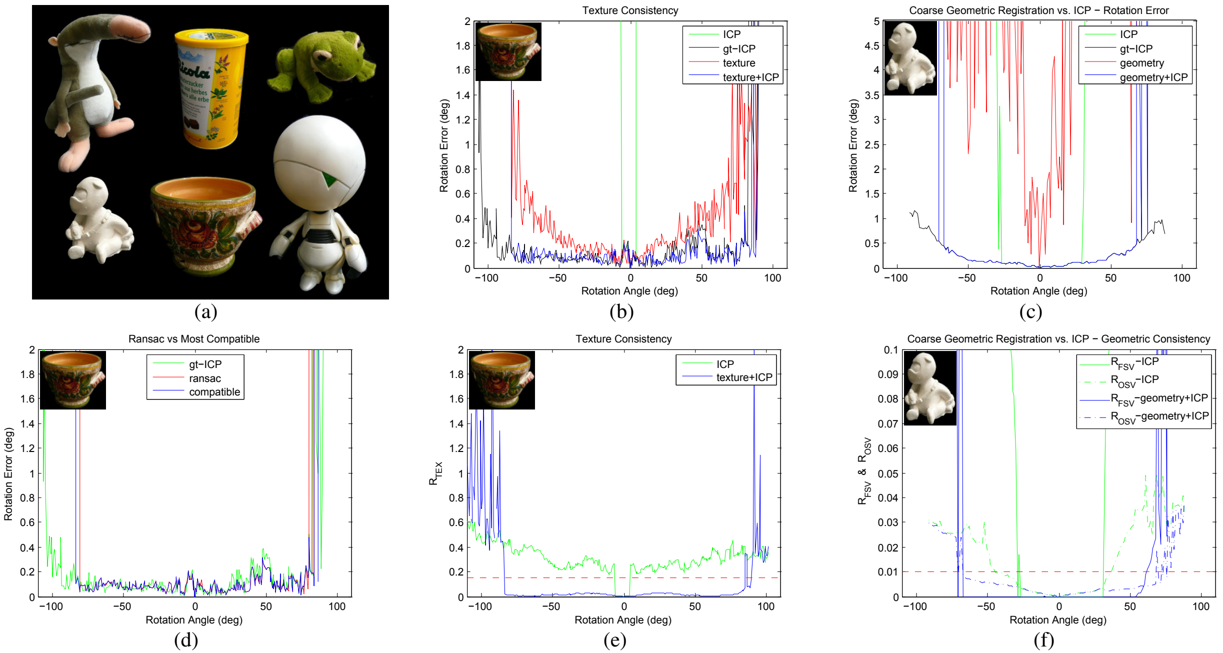


Figure 10. a) The test objects used for our experiments: *coati*, *ricola*, *frog*, *boris*, *pot*, *marvin*. b+e) Coarse texture registration for *pot*: ICP fails after a few degrees, but coarse texture registration is successful for up to 80°. The texture consistency ratio R_{TEX} shown in e) clearly indicates registration failure. c+f) Coarse geometry registration for *boris*: coarse geometric registration with ICP (up to 70°) clearly outperforms standard ICP (up to 30°). The geometric consistency ratios shown in f) increase considerably when registration fails, which allows us to detect these failures. d) Coarse texture registration for *pot*: Hypothesis generation based on most compatible feature pairs is as good as RANSAC with 50,000 hypotheses (using texture features + ICP). *(This figure is best seen in color.)*

[4] P. J. Besl and N. D. McKay. A method for registration of 3-D shapes. *PAMI*, 14:239–258, 1992.

[5] R. C. Bolles and P. Horaud. 3DPO: A three-dimensional part orientation system. *IJRR*, 5:3–26, 1986.

[6] R. J. Campbell and P. J. Flynn. A survey of free-form object representation and recognition techniques. *CVIU*, 81, 2001.

[7] Y. Chen and G. Medioni. Object modelling by registration of multiple range images. *IVC*, 10:145–155, 1992.

[8] M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *CACM*, 24:381–395, 1981.

[9] N. Gelfand, N. J. Mitra, L. J. Guibas, and H. Pottmann. Robust global registration. In *Symp. Geom. Proc.*, 2005.

[10] N. Gelfand, S. Rusinkiewicz, L. Ikemoto, and M. Levoy. Geometrically stable sampling for the ICP algorithm. In *3DIM*, 2003.

[11] C. Harris and M. Stephens. A combined corner and edge detection. In *Proc. of Alvey Vision Conf.*, 1988.

[12] D. F. Huber and M. Hebert. Fully automatic registration of multiple 3D data sets. *IVC*, 21:637–650, 2003.

[13] T. Jaeggli, T. Koninckx, and L. V. Gool. Online 3d acquisition and model integration. In *Pro. Cam.*, 2003.

[14] A. E. Johnson and M. Hebert. Using spin images for efficient object recognition in cluttered 3D scenes. *PAMI*, 21, 1999.

[15] M. J. Jones and J. M. Rehg. Statistical color models with application to skin detection. *IJCV*, 46:81–96, 2002.

[16] S. B. Kang and A. E. Johnson. Registration and integration of textured 3-D data. In *3DIM*, 1997.

[17] X. Li and I. Guskov. Multiscale features for approximate alignment of point-based surfaces. In *SGP*, 2005.

[18] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60:91–110, 2004.

[19] S. Malassiotis and M. G. Strintzis. Snapshots: A novel local surface descriptor and matching algorithm for robust 3D surface alignment. *PAMI*, 29:1285–1290, 2007.

[20] P. Merrell, A. Akbarzadeh, L. Wang, P. Mordohai, J.-M. Frahm, R. Yang, D. Nister, and M. Pollefeys. Fast visibility-based fusion of depth maps. In *ICCV'07*, 2007.

[21] R. Mester, T. Aach, and L. Dümbgen. Illumination-invariant change detection using a statistical colinearity criterion. In *DAGM*, 2001.

[22] S. Rusinkiewicz, O. A. Hall-Holt, and M. Levoy. Real-time 3D model acquisition. *Siggraph*, 21:438–446, 2002.

[23] S. Rusinkiewicz and M. Levoy. Efficient variants of the ICP algorithm. In *3DIM'01*, 2001.

[24] J. K. Seo, G. C. Sharp, and S. W. Lee. Range data registration using photometric features. In *CVPR*, 2005.

[25] M. K. Szymon Rusinkiewicz, Benedict Brown. 3d scan matching and registration. ICCV'05 Short Course Notes. http://www.cs.princeton.edu/b̃jbrown/iccv05_course/.

[26] D. Tubic, P. Hébert, J.-D. Deschênes, and D. Laurendeau. A unified representation for interactive 3D modeling. In *3DPVT*, 2004.

[27] T. Weise, B. Leibe, and L. V. Gool. Fast 3d scanning with automatic motion compensation. In *CVPR'07*, 2007.

[28] S. Zhang and P. Huang. High-resolution, real-time 3d shape acquisition. In *CVPRW, Sensor3D*, 2004.