Robust Automatic Registration of Range Images with Reflectance

Marcel Körtgen[*]

Institute of Computer Graphics
University of Bonn / Germany

Abstract

We tackle the problem of automatic matching, consistency checking and registration of multiple unknown and unordered range images. Utilizing robust visual features extracted from the supplied reflectance images, an efficient pairwise view matching scheme is used to build up a directed correspondence graph, nodes representing the input range images and edges labeled with relative pose estimates. Subsequently, a local and global consistency check eliminate false positive edges in the graph as these prevent the succeeding to a correct solution. Absolute poses are recovered by a breadth-first search (BFS), thereby, for each visited node, combining the weighted contributions of all encountered paths back to the root. Remarkably, the absolute alignments are accurately recovered from only the features. Thus, a subsequent fine registration step can be omitted. The framework is independent from object size and particular sensor model.

Keywords: features, SIFT, surface matching, hypothesis testing, pose consistency, breadth first search

1 Introduction

Registration has been an active topic of research for about thirty years. Much work in the past successfully addressed pairwise registration, i.e. aligning two three-dimensional (3d) views of a static scene. Due to the ongoing advances in scanning and computer hardware for about the last ten years, multiple view registration became manageably and thus more and more attractive as a basic tool in reconstructing a complete 3d model from a captured scene. However, most multiview approaches assume that the input views are roughly prealigned or that it is known which views overlap one another. In contrast, just a small amount of research has been published that engages automatic matching, consistency checking and registration of multiple unknown views as presented here.

Today, a laser range scanner is the method of choice for digitizing real-world objects of moderate size. Laser range scanners are non-contact 3d scanners that measure the distance from the sensor to points in the scene, typically in a regular grid pattern. A *range image* is the visualization of this grid pattern where the pixel intensity is a function

Figure 1: Example range image and co-registered reflectance image obtained with the Minolta Vivid 900.

of the measured distance (cf. figure 1). A natural byproduct of the acquisition process is the *reflectance image* that records the laser reflectance strength (LRS) for each pixel.

In general, because of occlusion and field of view limitations, not all parts of the scene can be observed from any given position. Therefore, range data from multiple viewpoints must be combined to form a complete model of the scene. Given a set of *n* overlapping range images of a static scene, the process of creating the complete scene model consists of two main steps: *registration* and *reconstruction*. In the registration step, the *n* input range images are all aligned in a common coordinate system whereas the reconstruction step usually accounts for the generation of a triangulated mesh out of the registered range data [1, 2].

The outline of the rest of the paper is as follows: The next section reviews related work. In section 3 we give an overview of the proposed pairwise view matching. Each pairwise match is introduced into the model graph as an edge. Edge and graph consistency checking are described in section 4 whereas absolute pose recovery is considered in section 5. An application result is shown in section 6 and conclusions are given in section 7.

---

[*]koertgen@cs.uni-bonn.de

2 Related Work

The first and most prominent approach to pairwise registration ($n = 2$) is the *Iterative Closest Points* (ICP) [3] algorithm, which has numerous variants [4]. However, ICP and its variants usually require the input to be roughly prealigned. If not the case point correspondences need to be found by using more sophisticated techniques, referred to as *surface matching* [5] or *surface correspondence* [6]. In such approaches, point correspondences are typically found by matching additional distinctive properties like geometrical or color features [7, 8, 6]. Often, the registration process is also split up into a crude alignment step (based on the features) and a fine registration step (usually ICP) [9, 10, 11, 12, 13, 14].

For *unordered* input, it is unknown which of the $n > 2$ scans are neighbouring or overlapping. It is easy to extend an existing surface matching algorithm to a *scan matching* algorithm [15] by simply applying it for all possible $O(n^2)$ pairs. Therefore, in analogy to information retrieval and hypothesis testing, we call it a *matching hypothesis* when a surface matching algorithm succeeds in finding sufficient correspondences between two scans. However, a matching hypothesis is not enough because any scan matching algorithm can easily be fooled by similar structures appearing repeatedly at different locations in the input data (see figure 2). Because accepting an incorrect hypothesis drastically reduces the chance of finding the correct overall solution, it is thereafter necessary to perform a consistency test. In the context of full- and semi-automated registration this is referred to *consistency checking*, *verification* or *classification* [16, 5, 17]. An incorrect but accepted hypothesis is named *false positive* whereas a correct but rejected hypothesis is named a *false negative*. In summary, the registration procedure for unordered input typically reads as follows:

1. **Scan matching**
   Perform pairwise surface matching on all $O(n^2)$ pairs of input range images, produce matching hypotheses and obtain crude alignment by least squares.

2. **Consistency checking**
   Perform a binary classification (accept/reject) for the found matches and delete the rejected matches.

3. **Refinement**
   Refine the relative transformations (usually ICP-based).

4. **Absolute Pose**
   Choose a common coordinate basis and rewrite the relative poses according to this basis.

For step 4 (absolute pose), it is helpful to consider a directed graph $G = (V, E)$ where the node set $V := \{V_i\}$; $i = 1, \ldots, n$ represents the n input range images and the edge set $E = \{e_i^j = (V_i, V_j)\}$ represents the matching hypotheses [5, 10, 11]. If G is connected, there exists a path

Figure 2: Motivational example for consistency checking. A scan matching algorithm will probably matching the highlighted region (red), although this obviously incorrect.
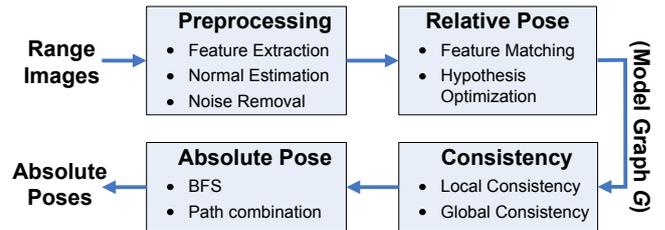
Figure 3: The Registration Pipeline.

$\pi_k^l = (V_k, \cdots, V_l)$ connecting any two vertices V_k and V_l. So, when selecting V_0 as the root node, the pose of any other node V_i can be obtained with respect to V_0 by accumulating the relative poses along the path π_0^i. In the works cited above, a minimum spanning tree (MST) of G is used to avoid long paths for numeric reasons. However, since the MST is only a minimum subgraph, this means ignoring most of the available information.

We make a last distinction for the case of *unknown input*, which means that no specific assumptions are made on the input, such as the presence of markers [18] or that the captured scene belongs to a distinct and known category (e.g. cars, indoor, single object, etc.). In the input category was known, empirical thresholds derived from representative training data could be used for decision making [19, 20, 21, 22, 23, 24, 9, 17]. Also, supervised learning techniques like bayesian classifiers [5] or support vector machines could be used. However, when dealing with unknown input, this does not apply because of missing training data.

3 Relative Pose

Figure 3 gives a high-level overview of the proposed registration framework. Input are n unknown and unordered range images $\{V_i\}$. If not already present, the preprocessing stage encompasses normal estimation by gaussian weighted plane fitting to local k-neighbourhoods [16]. Noise caused by the acquisition device, is removed by novel yet simple outlier rejection scheme [16] that exploits the regular grid structure of the range image.

Searching for point correspondences is speeded up by using distinctive features that are able to guide the search
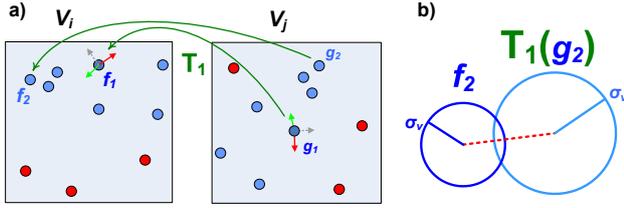
Figure 4: Hypothesis Evaluation for the view pair (V_1, V_2). (a) A pose hypothesis T_1 is generated from the candidate match (f^1, g^1) (blue) and evaluated by counting inliers (light blue). (b) Another match (f^2, g^2) is accepted as an inlier if mapped closer than the sum of the positional uncertainties σ_v.
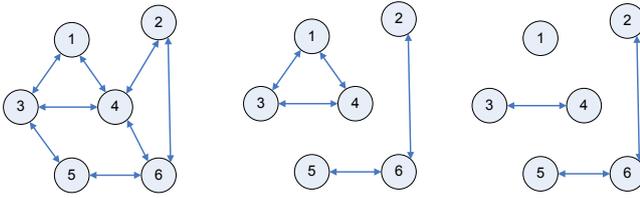
Figure 5: Example model graphs. A complete model graph (left), two partial models (middle) and an example with an isolated node (right).

early into the most likely direction. We apply the *Scale Invariant Feature Transform* (SIFT) [25] on the reflectance images which encode the material information of the captured surfaces. Thus, a matching of the extracted features is more likely to result in unique point correspondences, than geometric matching alone. The SIFT descriptor provides image location, orientation and scale and was recently shown to be highly distinctive [26]. Each feature is annotated a 3d position v along with a corresponding uncertainty estimate σ_v, both computed by robust maximum-likelihood estimation [16]. Moreover, each feature is assigned a local 3d basis constructed by Gram-Schmidt orthonormalization using the feature's orientation in the image plane and the estimated surface normal. The local basis is the key for efficient relative pose estimation. It is performed for all $O(n^2)$ pairs of input views and can be summarized as follows:

1. **Selection of potential feature matches**
   For each SIFT feature extracted from the first view, the nearest neighbour from the second view is found. This is speeded up by approximate nearest neighbour (ANN) search in the high-dimensional feature space similar to the Best-Bin-First (BBF) algorithm proposed in [27]. Ambiguous nearest-neighbours are rejected by using the Closest-To-Next-Closest (CTNC) [25].

2. **Hypothesis optimization**
   The least squares transformation from one coordinate frame to another can be computed in closed-form re-

quiring a minimum of three point correspondences [28]. Note that these are given by the feature's local 3d basis vectors so that a pose hypothesis can be generated from only a single feature match. For each candidate match, we count the number of inliers supporting the hypothesized pose, as depicted in figure 4. Finally, the transformation for the best hypothesis is refined by computing the least squares transformation for all inliers. For further implementation details we refer to [16].

The output of the relative pose step is a model graph (see figure 5). A model graph is an undirected Graph $G = (V, E)$ where the nodes $V = \{V_i\}$ represent the n input range images. Pairs of matched views are connected by an edge $e_i^j = (V_i, V_j)$. Associated to each node V_i is an absolute pose T_i and to each edge e_i^j a relative pose T_i^j as well as various other registration attributes.

4 Consistency checking

We propose a novel two-stage consistency check. First, all edges are tested to be *locally consistent*, which means that both views agree on the on the mutually seen volume. Second, a concept of *global consistency* (section 4.2) is derived similar to [9] where the observation is explored that accumulating the relative transformations along a cycle path in the graph should result in the identity transformation.

4.1 Local consistency

The proposed local consistency check can be seen as a hypothesis test, which attempts to refute one claim H_0 in favour of the complementary claim H_1 based on observation data $\{\tau_k\}_k$. Informally, these two claims are:

- H_0 - "e_i^j is locally consistent" and

- H_1 - "e_i^j is not locally consistent, and thus incorrect".

For the sample data $\{\tau_k\}_k$ we consider the euclidean 3d distances between corresponding surface points of the two views V_i and V_j. To evaluate these distances we perform bilateral image warping similar to [29]. For ease of notation, let V_j' denote the view V_j warped to the viewpoint of V_i and V_i' the view V_i warped to V_j. Now, evaluating the distances of corresponding surface points reduces to computing two difference images $\Delta_{V_i, V_j'}$ and $\Delta_{V_j, V_i'}$. For a correct match, the distances $\Delta_{V_i, V_j'}(x_k, y_k)$ and $\Delta_{V_j, V_i}^{\Omega}(x_l, y_l)$ should be small for any defined raster points $(x_k, y_k), (x_l, y_l)$. By contrast, a large observed distance should not be forejudged to indicate an inconsistency unless aspects of visibility, occlusion or device limitations can be ruled out. Moreover, it should be considered what, if anything, can be concluded for the surface consistency when one of the corresponding pixels is not defined.

4.1.1 Space violations with respect to visibility

Following the notation of [16, 5], figure 6 depicts the few basic situations to arise

Surfaces are close   If distances between corresponding pixels are reasonably small, the surfaces can be considered consistent. However, instead of thresholding we assign a likelihood value to each observed distance (cf. section 4.1.2). Therefore, the separation to the following cases should be considered for conceptual reasons only.

Free/Blocked Space Violation (FSV/BSV)   The FSV has recently been introduced by [5]. The warped surface $V_j'(x,y)$ blocks the visibility of the source surface $V_i(x,y)$. This violates the assumption of the source view V_i that the space along the ray of sight should be *free* until it reaches the surface $V_i(x,y)$. Since V_i has the most correct visibility information w.r.t to its own viewpoint, this is an unmistakable inconsistency. The BSV is the counterpart to the FSV; both are considered in figure 6 (b).

Empty Space Violation (ESV)   An ESV occurs when one of the two pixel operands is not defined. It violates the assumption of one view, that the space along the line of sight should be *empty* whereas the other view detected a solid surface. For more details on space violations we refer the reader to [16].

If we used an ideal range scanner that never misses a solid surface no matter how specular, how dark or how far away it is, our violation model would be complete. In practice, however, laser range scanners are limited and they may fail at surface detection for various reasons. We therefore consider fuzzy violations by assigning observation confidences $\{\mu_k^O\}_k \in [0,1]$ as described in [16].

4.1.2 Probabilistic model and binary classification

For the local consistency measure P_i^j, we make the following assumptions [16]:

1. The decision whether H_0 applies or not, is totally determined by the mutually observed region O.

2. Observations on different pixels are independent.

3. The probability $P[\tau_k]$ can be approximated by the normalized observation confidence $\mu_k^O / \sum_l \mu_l^O$,

4. The probability $P[H_0|\tau_k]$ can be modeled by a Poisson process with rate $\lambda = 1/\sqrt{\text{mse}_i^j}$ where mse_i^j is the mean squared error of the 3d positions of the features matched between V_i and V_j.

The first two assumptions allow us to apply the law of total probability, i.e.

$$P[H_0] = \sum_k P[H_0|\tau_k] \cdot P[\tau_k]. \tag{1}$$

The third assumption says that $P[\tau_k]$ does not depend on the surface distance observed in the kth pixel but only on the observation confidence. A *Poisson process* with rate $\lambda > 0$ is an integer-valued, independent and stationary continuous time stochastic process $\{X(t); t \geq 0\}$ with the *Poisson distribution*

$$P[X(t+s) - X(s) = l] = \frac{\lambda^l \cdot e^{-\lambda t}}{l!}. \tag{2}$$

It is convenient to view the Poisson process $X(t)$ as a special counting process, where the number of events in any interval of length t is quantified by the Poisson distribution. For our case, the length t of the time interval corresponds to the observed distance τ_k and the number of events l equals the number of surface inconsistencies to be observed when H_0 applies, namely 0. Then, the Poisson distribution reduces to the exponential reliability function $e^{-\lambda t}$. By substituting and approximating (1) we obtain:

$$P[H_0] \approx \frac{1}{\sum_l \mu_l^O} \sum_k e^{-\lambda \tau_k} \cdot \mu_k^O =: P_i^j, \tag{3}$$

which can be seen as the confidence-weighted sample mean of the exponential reliability $e^{-\lambda t}$. In [16] the cut-value of 0.5 is motivated for binary classification, i.e. edges e_i^j with a value $P_i^j < 0.5$ are rejected as locally inconsistent. However, if an edge is accepted there might still be inconsistencies undetectable from the two considered views V_i, V_j. Therefore, a consistency concept that extends to the whole model graph G is introduced in the following section.

4.2 Global consistency

The basic idea is that in a correct graph each cycle is *pose consistent*. Consider a model graph G representing a correct registration. Let

$$(V_{\pi(0)}, V_{\pi(1)}, \cdots, V_{\pi(k-2)}, V_{\pi(k-1)}, V_{\pi(0)})$$

be a cycle path of length k and $\pi : [0 : k-1] \mapsto [0 : n-1]$ a reordering which just eases the notation. When composing the relative motions along the cycle, the resulting pose should equal the pose of the start node $V_{\pi(0)}$, i.e.

$$T_{\pi(0)}^{\pi(1)} \circ T_{\pi(1)}^{\pi(2)} \circ \cdots \circ T_{\pi(k-2)}^{\pi(k-1)} \circ T_{\pi(k-1)}^{\pi(0)} =: \hat{T} \approx \mathbb{I}. \tag{4}$$

In practice, the condition $\hat{T} = \mathbb{I}$ is never met due to noise in the data which in turn introduces errors in the transformation estimates. As a measure of cycle consistency, we consider the mean distortion that $\hat{T}$ introduces to $V_{\pi(0)}$ and compare this to the sum of edge errors in the whole cycle. If the sum is exceeded, we consider the cycle inconsistent.

The breadth-first-search BFS defines a spanning tree [1] of G and additionally, it tends to prefer short over long cycles and thus sharpens the cycle error bound. Moreover,

---

[1]Note that any spanning tree of an undirected connected graph spans the graph's cycle space [30]. This means that 1) by adding an edge to any spanning tree, a cycle is closed and 2) the set of cycles detected this way can generate all cycles in the graph by union and intersection.

a) consistent surfaces b) Free/Blocked space violation (FSV/BSV) c) Empty space violation (ESV)
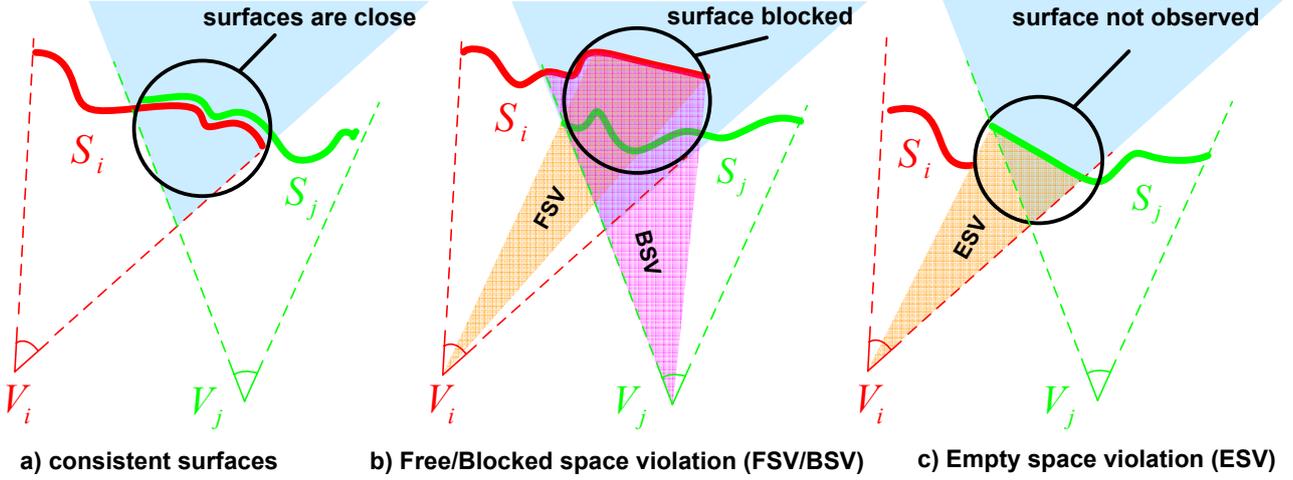
Figure 6: Local consistency by means of visibility. (a) consistent surfaces are close together wherever they overlap (blue region); (b) a *free* or *blocked space violation* occurs when surfaces block each other's visibility (highlighted regions); (c) an *empty space violation* occurs when a surface is not observed although expected to.
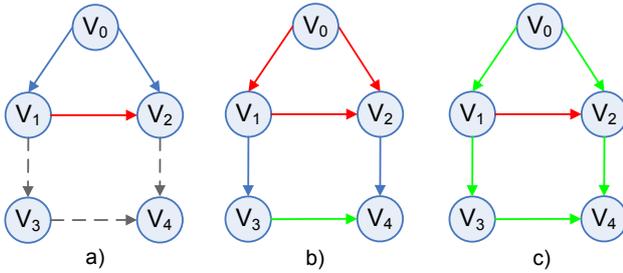
Figure 7: Global consistency through breadth-first traversal in the model graph G.

Figure 8: (a) In an MST-based approach the absolute poses are determined from one path (blue) to the root node whereas the additional information provided by indirect paths (red) is ignored; (b) Using a modified BFS takes indirect paths into account.

by maintaining the same traversal order for the absolute pose recovery, we make sure that only the relevant cycles are visited. A motivational example is given in figure 7. A breadth-first search is started from node V_0. The algorithm traverses the edges (V_0,V_1), (V_0,V_2) (blue). Moving on to V_1 and traversing the edge (V_1,V_2) an inconsistent cycle arises. (7 b). All edges of the inconsistent cycle are marked as **probably wrong** (red) and the traversal proceeds until a second cycle is closed at (V_3,V_4), this time it is pose consistent. Now all edges back to the root node can be labeled as **approved**(green), leaving the inconsistent edge (V_1,V_2). Finally note that, although checking pose consistency of cycles is a useful tool, it is not failsafe. For more details including the selection of root nodes we refer to [16].

5 Absolute Pose

The last step in the registration pipeline is to provide an absolute transformation for each node with respect to one common coordinate system. Figure 8 (b) suggests that the absolute node transformations can be estimated through

a weighted combination $T_i := \sum_j \omega_j \cdot T_j^*$ of the accumulated transformations T_j^* of all paths j that lead into node V_i. accumulation. However, linear interpolation of $n > 2$ rigid motions is difficult due to the non-linearity of rotation [31]. Instead we suggest to pick up the feature matches M_i^j when traversing the edge e_i^j and propagate them to the edge transformations. When arriving at a node we solve for the weighted least squares transformation of all previously collected and transformed feature matches. The traversal order is given by the same BFS as performed in the previous section except that promising paths over nodes of high degree and confidence are examined first. During traversal, parent nodes are tracked so that the path back to the root node is instantly available for any visited node. Also, feature matches of incoming edges are collected, transformed and stored at the respective node. When arriving at a distinct node, we examine all adjacent nodes that have been visited so far, thereby incrementally combining the feature matches of all direct and indirect

paths yet visited to solve for an absolute pose. More details are given in [16].

6 Results

The **church** model was automatically scanned as a sequence of small incremental $12°$ rotations using a turntable, resulting in 30 views. For one view, the object has been missed completely due to range miscalibration, so that only background clutter was measured. The view remained isolated, i.e. totally unmatched, and was therefore silently removed. The small rotation angle of $12°$ provided sufficient overlap for up to five rotations - for instance, the view V_{19} was correctly matched against the subsequent views $V_{20}, V_{21}, V_{22}, V_{23}$ and V_{24} - and thus eased view matching, resulting in a model graph with high connectivity. However, the workload for local consistency was also significant: 306 edges needed to be checked, from which all 227 incorrect edges were found. Timings were: $2 : 42m$ (preprocessing), $73.89s$ (matching), $226.65s$ (local consistency), $0.05s$ (global consistency) and $0.62s$ (absolute pose). Preprocessing includes conversion from proprietary file formats, SIFT feature extraction[2], normal estimation and noise removal. The model graph for the remaining 79 edges is shown in 9 (left). Note how the sequential turntable scanning order $(V_0, \ldots, V_{28})$ can be followed by the string-like shape of the graph. More results are presented in [16].

7 Summary and Conclusions

We presented a framework and some novel ideas for automatic registration of multiple unordered views of an unknown scene. To handle false positive edges, we conceived a novel test for both local and global consistency. Local consistency was engaged with respect to the visibility of both involved viewpoints and expressed in terms of the euclidean distances between all corresponding surface points. By using a bilateral image warping scheme, these distances could be evaluated basically in the form of a difference image. We introduced the concept of the ESV which extends the consistency reasoning from the overlapping region to the whole mutually seen region. With this, a local consistency measure was presented, based on 1) a fuzzy model to assess the confidence of a pixel observation and 2) the Poisson process to model the likelihood for the observed pixel surface distance under the H_0-assumption. Edge classification was then performed by thresholding the value of the local consistency measure against the cut-value of 0.5.

Global consistency is expressed in terms of pose consistent cycles in the model graph. A measure for the pose

consistency of a cycle was defined as the mean displacement of a view's surface points when propagated along the cycle. However, a concept of consistency cannot imply correctness. In the same way a locally consistent edge can still be incorrect, also a cycle can almost be perfectly pose consistent and yet still be incorrect. Finally, absolute poses were recovered by composing relative poses along the path to the root node. In contrast to common MST-approaches, we proposed not to rely on a single path but on a incremental, weighted combination of multiple paths back to the root node by means of a BFS-scheme. Thereby, we obtain more accuracy and avoid error accumulation as well as the subsequent costly fine registration step.

We conclude that the presented framework performs efficiently and accurately without user intervention. They keys for efficient registration as presented here are 1) to use the feature metric to quickly guide pairwise matching into the most likely direction, 2) to carry out optimization on features only and 3) to use a path traversal scheme in favor of global optimization. When registering on the order of some thousand views, the bottleneck will be the local consistency check which is by far the most expensive operation in the pipeline. To accomplish this, one might investigate in hierarchical data structures like range image pyramids, for instance. An example for a future application is guided scanning, where each shot is instantly registered and camera locations for subsequent shot are suggested by software.

References

[1] Nina Amenta, Sunghee Choi, and Ravi Krishna Kolluri. The power crust, unions of balls, and the medial axis transform. *Computational Geometry*, 19(2-3):127–153, 2001.

[2] T. Dey and S. Goswami. Tight cocone: A water-tight surface reconstructor, 2003.

[3] P. J. Besl and N. D. McKay. A method for registration of 3-D shapes. *IEEE Transactions on Pattern Analysis and machine Intelligence*, 14(2):239–258, February 1992.

[4] Szymon Rusinkiewicz and Marc Levoy. Efficient variants of the ICP algorithm. In *Proceedings of the Third Intl. Conf. on 3D Digital Imaging and Modeling*, pages 145–152, 2001.

[5] Daniel Huber. *Automatic Three-dimensional Modeling from Reality*. PhD thesis, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, December 2002.

[6] Peter Biber and Wolfgang Straßer. Solving the correspondence problem by finding unique features. In *16th International Conference on Vision Interface*, 2003.

---

[2]we used the authors implementation available at `http://www.cs.ubc.ca/~lowe/keypoints/`
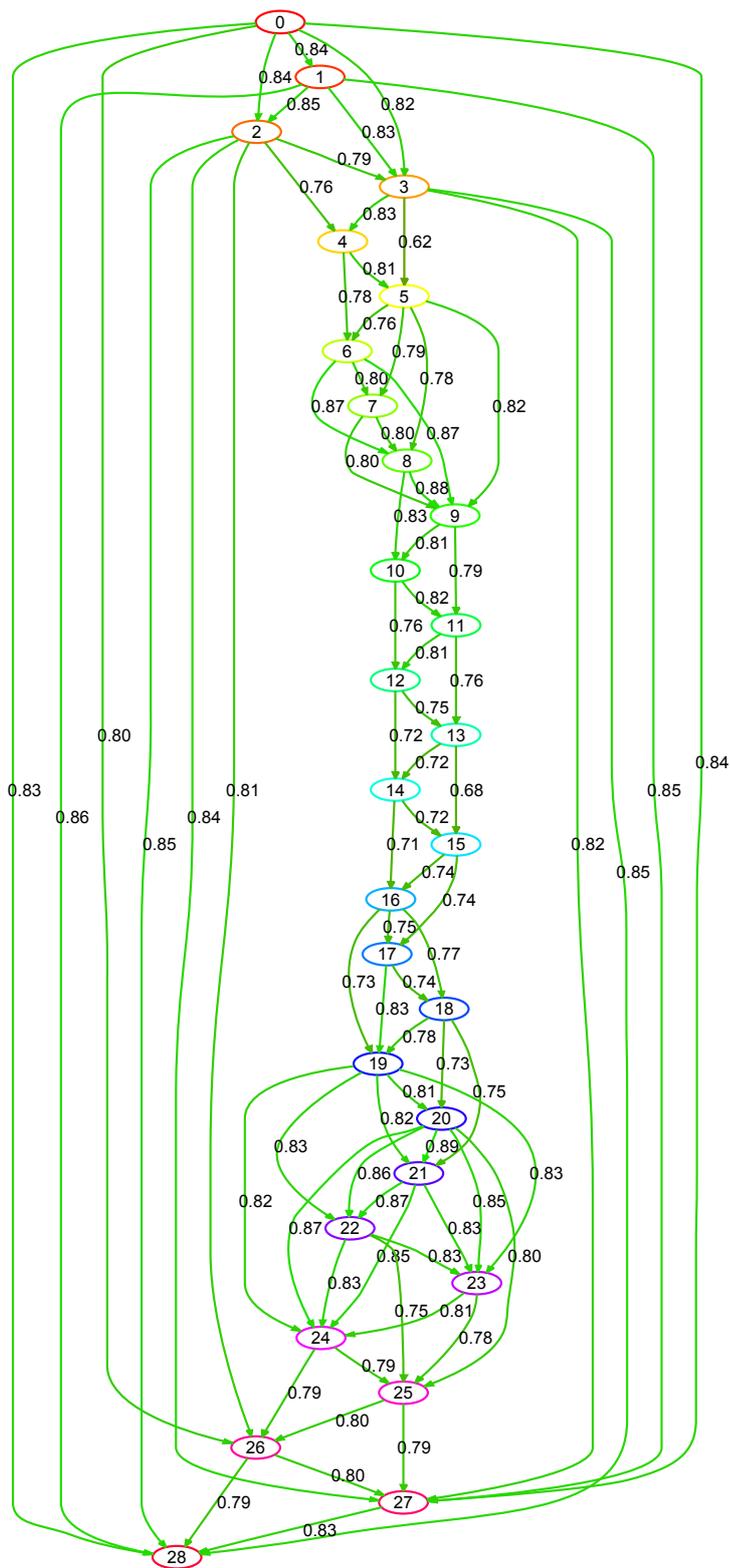
Figure 9: A complete solution for the **church** model with 29 views. The small rotation steps of 12° allowed to match views over wide index shifts up to 5 (e.g. $e_0^3, e_5^9, e_{19}^{24}, e_2^{27}, e_3^{26}$). Note that the graph is essentially shaped like a string, indicating the sequential adjacency that results from any turntable setup. (right) 3d point renderings of the solution; 1.637.000 points.

[7] S. Gumhold, X. Wang, and R. McLeod. Feature extraction from point clouds, 2001.

[8] M. Pauly, R. Keiser, and M. Gross. Multi-scale feature extraction on point-sampled surfaces, 2003.

[9] Gregory C. Sharp. *Automatic and Stable Multiview 3D Surface Registration*. PhD thesis, University of Michigan, Oct 2002.

[10] Gerhard H. Bendels, Patrick Degener, Roland Wahl, Marcel Körtgen, and Reinhard Klein. Image-based registration of 3d-range data using feature surface elements. In *VAST*, pages 115–124, 2004.

[11] Xinju Li and Igor Guskov. Multiscale features for approximate alignment of point-based surfaces. In *Symposium on Geometry Processing*, pages 217–226, 2005.

[12] N. Gelfand, N. J. Mitra, L. J. Guibas, and H. Pottmann. Robust global registration. In *Symposium on Geometry Processing*, pages 197–206, 2005.

[13] Shankar Krishnan, Pei Yean Lee, John B. Moore, and Suresh Venkatasubramanian. Global registration of multiple 3d point sets via optimization-on-a-manifold. In *Symposium on Geometry Processing*, pages 187–196, 2005.

[14] Ameesh Makadia, Alexander Patterson, and Kostas Daniilidis. Fully automatic registration of 3d point clouds. In *IEEE Conference on Computer Vision and Pattern Recognition,*, 2006.

[15] Peter Biber and Wolfgang Strasser. nscan-matching: Simultaneous matching of multiple scans and application to slam. In *IEEE International Conference on Robotics and Automation*, 2006.

[16] Marcel Körtgen. Robust automatic registration of range images with reflectance. Master's thesis, Computer Graphics Institute, University of Bonn, Germany, 2006. Available at http://marcel.koertgen.de/marcy/research/2006/da-koertgen-2006.pdf.

[17] Bradford J. King, Tomasz Malisiewicz, Charles V. Stewart, and Richard J. Radke. Registration of multiple range scans as a location recognition problem: Hypothesis generation, refinement and verification. In *3DIM*, pages 180–187, 2005.

[18] Devrim Akca. Full automatic registration of laser scanner point clouds. In A. Gruen and H. Kahmen, editors, *Optical 3-D Measurement Techniques VI*, volume 1, pages 330–337, 2003.

[19] Gerard Blais and Levine Martin. Registering multiview range data to create 3D computer objects. Technical Report CIM-93-16, McGill Centre for Intelligent Machines, March 1994.

[20] Xavier Pennec. Multiple Registration and Mean Rigid Shapes - Application to the 3D case. In I.L. Dryden K.V. Mardia, C.A. Gill, editor, *Image Fusion and Shape Variability Techniques (16th Leeds Annual Statistical (LASR) Workshop)*, pages 178–185, july 1996.

[21] David W. Eggert, Andrew W. Fitzgibbon, and Robert B. Fisher. Simultaneous registration of multiple range views for use in reverse engineering of cad models. *Comput. Vis. Image Underst.*, 69(3):253–272, 1998.

[22] Chitra Dorai, Gang Wang, Anil K. Jain, and Carolyn Mercer. Registration and integration of multiple object views for 3d model construction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(1):83–89, 1998.

[23] S.J. Cunnington and A.J. Stoddart. N-view point set registration: A comparison. In *British Machine Vision Conference*, volume 1, pages 234–244, Nottingham, UK, 1999.

[24] Kari Pulli. Multiview registration for large data sets. In *Proceedings of the International Conference on 3D Digital Imaging and Modeling*, pages 160–168, 1999.

[25] David Lowe. Distinctive image features from scale-invariant keypoints, 2003.

[26] Krystian Mikolajczyk and Cordelia Schmid. A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 27(10):1615–1630, 2005.

[27] J. Beis and D. Lowe. Shape indexing using approximate nearest-neighbor search in highdimensional spaces, 1997.

[28] Berthold K. P. Horn. Closed-form solution of absolute orientation using unit quaternions. *Journal of the Optical Society of America A*, 4(4):629–642, April 1987.

[29] Leonard McMillan. *An Image-Based Approach to Three-Dimensional Computer Graphics*. PhD thesis, University of North Carolina, Apr 1997.

[30] Reinhard Diestel. *Graph Theory*, volume 173. Springer-Verlag, Heidelberg, 3 edition, July 2005.

[31] Charles Bloom, Jonathan Blow, and Casey Muratori. Errors and omissions in marc alexas "linear combinations of transformations", 2004. Available at http://www.cbloom.com/3d/techdocs/lcot_errors.pdf.