Vision, Modeling, and Visualization (2014)
J. Bender, A. Kuijper, T. von Landesberger, H. Theisel and P. Urban (Eds.)

SOAR: Stochastic Optimization for Affine global point set Registration

M. Agus[1] E. Gobbetti[1] A. Jaspe Villanueva[1] C. Mura[2] R. Pajarola[2]

[1] Visual Computing, CRS4, Pula, Italy [2] Visualization and MultiMedia Lab, University of Zurich, Switzerland

Abstract

We introduce a stochastic algorithm for pairwise affine registration of partially overlapping 3D point clouds with unknown point correspondences. The algorithm recovers the globally optimal scale, rotation, and translation alignment parameters and is applicable in a variety of difficult settings, including very sparse, noisy, and outlier-ridden datasets that do not permit the computation of local descriptors. The technique is based on a stochastic approach for the global optimization of an alignment error function robust to noise and resistant to outliers. At each optimization step, it alternates between stochastically visiting a generalized BSP-tree representation of the current solution landscape to select a promising transformation, finding point-to-point correspondences using a GPU-accelerated technique, and incorporating new error values in the BSP tree. In contrast to previous work, instead of simply constructing the tree by guided random sampling, we exploit the problem structure through a low-cost local minimization process based on analytically solving absolute orientation problems using the current correspondences. We demonstrate the quality and performance of our method on a variety of large point sets with different scales, resolutions, and noise characteristics.

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling —Geometric algorithms, languages, and systems

1. Introduction

Pairwise 3D point set registration is a fundamental computational geometry problem. It consists of finding the spatial transformation T that aligns two point sets. The problem is heavily studied and many solutions, tuned to particular situations, have been proposed (see Section 2). In this work, we look at the affine registration problem in which the transformation T has the form $T(\mathbf{q}) = sR\mathbf{q} + \mathbf{t}$, where s is a scaling factor, R a rotation matrix, and $\mathbf{t}$ a translation vector. Rigid registration, where $s = 1$, is a subset of this problem. While very efficient (and sometimes optimal) rigid and affine solutions exist for particular simplified settings (e.g., [MB03, LH07]), the problem is much more complex when the two datasets are very sparse, noisy, and with unknown correspondences and amount of overlap. This situation arises, for instance, in cross-modality mapping (e.g, as a sub-problem of 2D-3D registration, when mapping sparse point clouds derived from structure-from-motion to denser clouds from range sensors [PGC11b, CDG*13]). This setting, for instance, makes finding reliable correspondences hard and does not permit using local descriptors for guiding the matching process. For this reason, approximate methods based on randomized heuristics currently offer the best performance [AMCO08, PB11].

Our approach. The technique is based on a stochastic approach for the global optimization of an alignment error function robust to noise and resistant to outliers. This error function is based on assigning a small number K of model correspondences to each of the data points, and on using the L_m norm (with $m \leq 1$) to evaluate the alignment error of each of the correspondences. At each optimization step, the method alternates between stochastically visiting a generalized BSP-tree representation of the current solution landscape to select a promising transformation, finding point-to-point correspondences using a GPU-accelerated technique, and incorporating new error values in the BSP tree. In contrast to previous work, instead of simply constructing the tree by guided random sampling, we exploit the problem structure through a low-cost local minimization process based on analytically solving absolute orientation problems using the current correspondences.

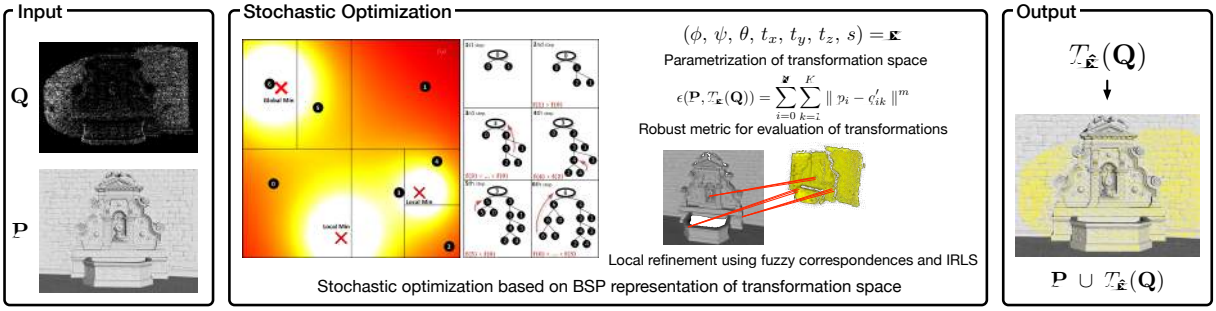


Figure 1: ***Our global registration pipeline:*** *Given as input two (unregistered) point clouds* **P** *and* **Q**, *we use a stochastic global optimization approach to find a transformation* $T_{\hat{\mathbf{x}}}$ *that aligns* **Q** *to* **P**.

Contribution. Our approach combines and extends state-of-the-art results in a general registration technique that does not require descriptors and can handle difficult cases, including affine registration of sparse noisy point clouds coming from multimodal acquisitions. Our method improves over recent work on stochastic optimization [PB11] in several ways: first of all, we extend the approach to affine registration; secondly, we robustify it using sparsity-inducing norms and one-to-many correspondences; finally, we improve the efficiency of the BSP exploration through guided splits and by exploiting domain knowledge to find local good solutions according to the robust error masure. We demonstrate the quality and performance of our method on a variety of large real-world point sets with different scales, resolutions, and noise characteristics.

Advantages and limitations. By incorporating domain knowledge into a stochastic global optimizer, we obtain a robust method that gracefully combines global domain exploration with local optimization. Sparsity inducing norms and fuzzy correspondence assignments make the method more robust without requiring data-dependent tuning to prune or down-weigh low quality correspondences. The proposed method also has a few limitations. As for other stochastic methods, long running times are sometimes required to reach the global optimum. Typical difficult cases for the method, shared with all other related methods working directly on points, occur when the objects exhibit strong slippage, leading to flat error landscapes. As the method is specifically aimed at difficult registration cases not handled by current solutions, other methods based on descriptors are typically more efficient when matched objects are dense and clean.

2. Related work

The problem of point set registration has been widely studied and a full review is out of the scope of this paper. We will restrict our analysis to methods for registering point sets using rigid or affine transformations (rigid plus scaling factor), which are close to our technique. Readers are referred to the work by Tam et al. [TCL*13] for a more complete survey. Our work is aimed at finding the alignment between two point clouds when a starting guess for the transformation is not given and without any knowledge about the mapping of the points. Classic pose estimation techniques use the generalized Hough transform [HB94], geometric hashing [WR97] and pose clustering [Sto87] voting schemes to find the optimal alignment (at least in the rigid case), but are limited to very small (and noise-free) or subsampled point clouds because of their computational complexity and memory footprint, while the Iterative Closest Point (ICP) algorithm [GP02] is the gold standard for local alignment tasks, i.e., when a rough alignment already exists. Recently, Bouaziz et al. [BTP13] proposed a new formulation of the ICP algorithm exploiting sparsity-inducing norms, which showed to be effective for rigid registration when dealing with outliers and partial overlap. We also considered L_m sparsity-inducing norms, but we enforced our distance metric by considering fuzzy assignment involving one-to-many mapping between the points. More complicated approaches for point cloud registration rely on local geometric descriptors, such as spin images [JH99] or integral volume descriptors [GMGP05], which are detected in both clouds and then matched. However, these methods suffer when datasets are infested by a large number of outliers or in the presence of strong asymmetry between the point clouds to be registered. To solve the issue of outliers, the 4PCS method [AMCO08] achieves robustness by combining a non-local descriptor (four coplanar points) with a generate-and-test RANSAC scheme. Recently, Mellado and colleagues [MMA14] improved the efficiency of the method, lowering its time complexity from quadratic to linear. Corsini et al. [CDG*13] extended 4PCS to estimate different scales between the point clouds to align. This approach, however, requires a partitioning into planar regions of the two point clouds, which is achieved through variational shape approximation [CSAD04], and could fail if one of the point clouds is sparse. To overcome this problem, the sparse input cloud is oversampled to a density comparable to the model point cloud. Another kind of registration methods employ robust statistics, such as general maximum likelihood estimation [GP02], kernel correlation [TK04] or mixture of gaussians [JV05]. In-

stead of using one-to-one correspondences, these approaches work with multiple, weighted correspondences. Although this significantly widens the basin of convergence, the computational cost limits the applicability to very small point clouds (hundreds of samples) [TARK10]. As an alternative to combinatorial optimization based on feature matching, some methods in the last decade tried to solve the pose estimation problem by minimizing a cost function with a global optimizer. For small datasets, rigid alignment algorithms based on deterministic branch-and-bound techniques [MB03, OKO09] or Lipschitz global optimization [LH07] have been proposed. Papazov and Burschka [PB11] recently introduced a stochastic global optimization approach for robust rigid point set registration, based on Bilbro and Snyder's tree annealing algorithm [BS91]. It is a stochastic sampling method that uses a generalized BSP tree and allows to minimize nonlinear scalar fields over complex shaped search spaces like the space of rotations. As a result, the method is robust and outlier-resistant. More recently, Pintus and Gobbetti [PG14] extended this approach to handle affine registration in the context of 2D-3D registration, employing metrics based on robust statistics [HR09]. In our work, we extend the stochastic approach to similarity transforms and combine it with a more domain-specific method that locally solves absolute orientation problems. Outliers are handled by employing a robust and adaptive pruning strategy and by using Iteratively Re-weighted Least Squares (IRLS) to solve absolute orientation tasks. This approach is feasible by harnessing the power of GPUs to rapidly compute correspondences between the two point clouds [Cay10]. Our error metric also benefits from sparsity-inducing norms and fuzzy assignments, which robustify the method without requiring data-dependent tuning to prune or downweigh low quality correspondences.

3. Method

Problem definition. Given two point clouds, referred as data point cloud $\mathbf{Q}$ and model point cloud $\mathbf{P}$, the alignment of the first with the latter requires the solution of a global optimization problem, consisting of determining the optimal affine transformation T of the form $T(\mathbf{q}) = sR\mathbf{q} + \mathbf{t}$ for a scaling factor s, a rotation matrix R and a translation vector $\mathbf{t}$, such that the point set $T(\mathbf{Q}) = \{T(\mathbf{q}_1), ..., T(\mathbf{q}_N)\}$ is best aligned with $\mathbf{P}$ according to some error criterion. By parameterizing the transformation T with a parameter vector $\mathbf{x}$, and defining an error function $\varepsilon(x)$ which determines the distance between P and $T_{\mathbf{x}}(\mathbf{Q})$, the optimal registration is given by minimizing $\varepsilon(x)$ over $\mathbf{x}$. We find the solution of the optimization problem through a problem-aware stochastic search of the parameter space (see Figure 1).

Parametrization. Our parameter vector $\mathbf{x}$ has 7 parameters: 1 scalar for the scaling factor s, 3 for the translation vector, t, and 3 for the three angles $(\phi, \psi, \theta) \in [0, 2\pi) \times [0, \pi] \times [0, \pi]$ necessary to define a redundant-free rotation space parametrization based on the axis-angle representation of $SO(3)$ (i.e., ϕ and ψ for the spherical coordinate representation of the rotation axis, and θ for the rotation amount around this axis). This rotation parametrization leads to simple techniques for achieving uniform sampling and equal volume bisection of the parameter space, and has already been used in other stochastic optimizers in the rotation space. For further details about space parametrization, search space structure and advantages in using this representation, see Section 4.1 in Papazov and Burschka [PB11] for a use closely related to our proposed method, and see Kanatani's book [Kan90] for a generic treatment of the topic. Our approach extends these methods by just adding the scale dimension.

Parameter range estimation. Given the point clouds $\mathbf{P}$ and $\mathbf{Q}$, it is trivial to determine the bounds for translation and rotation parameters that ensure that at least a minimal overlap between the point clouds exists. Setting bounds for scaling, however, requires some knowledge, e.g., to avoid the trivial solution of a null scaling factor and to restrict the search range. We cannot assume that the two point sets have perfect overlap, since the data point cloud could contain many points belonging to the model, but also a possibly large number of outliers from the surroundings. These points lead to potential problems in the estimation of the feasible scale range for the stochastic global optimization routine. Thus, in order to automatically determine a good scaling range, we use a heuristic to extract from the moving point cloud a subset of the object of interest. Similarly to other generic registration methods [AMCO08], we determine this information using bounds on the amount of overlap, namely we set as default limits for scaling 0.1-10 times the ratio between the bounding spheres of the point clouds.

Robust distance. The global optimization problem consists of minimizing an error function representing the distance between the model point cloud P and the transformed data point cloud $T_{\mathbf{x}}(\mathbf{Q})$. Since normally no a-priori information about the point sets is available, typical distance functions are not able to produce correct alignment of point clouds affected by noise or outliers. Hence, a way to reduce the effect of outliers is to consider robust regression methods, employing distance functions $\rho(x)$ which reduce the effect of outliers and noise by reducing the sample contribution to the error when the distance increases [HR09]. Using a generic robust kernel function $\rho(x, y)$, according to the current affine transformation $T_{\mathbf{x}}$, a generic error function takes the following form:

$$\varepsilon(\mathbf{P}, T_{\mathbf{x}}(\mathbf{Q})) = \sum_{i=1}^{N} \rho(p_i, T_{\mathbf{x}}(\mathbf{Q})). \quad (1)$$

Previous stochastic methods [PB11] considered various kind of robust estimators, giving good results for rigid registration, but depending on constants that are difficult to tune, and give different alignment quality according to the characteristics of point clouds considered. For example, Tukey bisquare and Huber estimators [HR09] are normally tuned according

to the standard deviation of errors, and they generally offer protection against uniform noise, but not against outliers or non-uniform noise.

Recently, Bouaziz et al. [BTP13] studied the characteristics of L_m norms in the context of generalized ICP, and they showed that choosing $m < 1$ significantly improves the resilience of the distance function to large amount of outliers, and to different kind of noise.

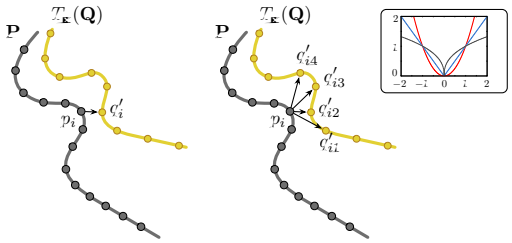


Figure 2: ***Fuzzy correspondences for error metric:*** *instead of mapping each point p_i of the model $\mathbf{P}$ to a single point of $T_{\mathbf{x}}(\mathbf{Q})$ (left), we assign it to its k nearest neighbors in $T_{\mathbf{x}}(\mathbf{Q})$ (right, with $k = 4$). To make our error metric more robust, we weight point-to-point distances using the function $|x|^{0.4}$ (gray curve in the inset), which gives outliers a smaller weights as compared to the more traditional x^2 (red curve) and $|x|$ (blue curve) penalty functions.*

To improve the robustness of the error metric, we employ fuzzy assignments between the point clouds: instead of one-to-one correspondences between data samples and model samples, we considered one-to-many mappings (see Figure 2). Specifically, each point in the model cloud is associated to the K nearest neighbors in the data cloud. Thus, the error metric does not use point-to-point or point-to-plane distances, but more robust fuzzy point-to-region distances, and our error function takes the following form:

$$\varepsilon(\mathbf{P}, T_{\mathbf{x}}(\mathbf{Q})) = \sum_{i=0}^{N} \rho(p_i, T_{\mathbf{x}}(\mathbf{Q})) = \sum_{i=0}^{N} (\sum_{k=1}^{K} \| p_i - q'_{ik} \|^m) \quad (2)$$

where $\{q'_{i1} = T_{\mathbf{x}}(q_{i1}), ..., q'_{iK} = T_{\mathbf{x}}(q_{iK})\}$ are the K nearest points of the data sample p_i. In our implementation, we considered $m = 0.4$ as exponent for the distance function, as suggested in [BTP13].

Stochastic optimization process. The optimal affine transform T is found with a method that effectively combines a stochastic exploration of the multidimensional parameter space with a local improvement scheme. Similarly to the purely stochastic method of Papazov and Burschka [PB11], we use a generalized BSP to represent our 7-dimensional search space and exploit it to adaptively add more detail to promising regions in the search space. For each step of the global optimization process, a node is added to the tree by splitting the parameter space in portions having the same volume (see Papazov and Burschka [PB11] for a motivation of this approach). Each node of the binary space partition tree contains the number of times it has been visited, the parameter bounds, the best parameter value and the value of the cost function evaluated in the subtree rooted at that node. At the end of the optimization procedure, the parameter values associated to the root are taken as solution. A branch-and-bound scheme is used to traverse the tree; a temperature with cubical decay ($t = \tau(1 - \frac{i}{i_M})^3$, with i the current iteration, i_M the maximum iteration count, and τ the initial temperature) is used to drive the probability p_l to choose the child associated with the lower function value. At the beginning of the optimization both branches have the same probability to be explored ($p_l = 0.5$), while at the end of the optimization the traversal is driven towards the branches with lower values ($p_l = 1.0$). Furthermore, in order to uniformly explore the parameter space and to reduce the chance of being trapped in local minima, the probability is weighted with the number of times in which branches have been traversed. Specifically, indicated with c_l and c_h the number of times the children associated with the lower and higher function value have been traversed, the probability p_l is computed as follows:

$$p_l = \frac{1 + t\,\frac{c_h}{c_l + c_h}}{1 + t}. \quad (3)$$

Moreover, a significant improvement in the cooling schedule is obtained by decomposing the main loop in different optimization loops. As the optimization level progresses, the number of iterations grows exponentially with base 2, while keeping trace of the previous optimization history by maintaining the tree status.

Local refinement through iteratively reweighted least squares. Our procedure, similarly to other alignment procedures, alternates between finding correspondences given a parameter value, evaluating the error using those correspondences, and using the result to move to other more promising parameter values. Since the most costly step is finding correspondences, we accelerate it through an approximate GPU-accelerated method [Cay10], and exploit them not just for error evaluation but also for error minimization. Given the correspondences, we obtain the minimum of ε using an iteratively reweighted least-squares (IRLS) [HW77] modification of Horn's method for solving the absolute orientation with Euclidean distances [Hor87], which consists in finding the optimal relationship, in a least-squares sense, between two coordinate systems using pairs of measurements of the coordinates of a number of points in both systems. In our case this is achieved by calculating weights $w_i = \varepsilon'(r_i)/r_i$, where r_i is the current residual and $\varepsilon'(d)$ is the derivative of $\varepsilon(d)$ (see Equation 2), by solving the weighted least squares problem

$$\mathbf{x} = \underset{\mathbf{x}}{\operatorname{argmin}} \sum_{i=0}^{N} \sum_{k=1}^{K} (w_k \| p_i - q'_{ik} \|^2). \quad (4)$$

using the analytical absolute orientation method of Horn, and by re-solving again until convergence. Since this process is included in a higher level global optimization method, we

just iterate for a fixed number of times (four in this paper) to find the (approximately) optimal solution in terms of robust distance given the current correspondences. This solution is then incorporated in the current error landscape by inserting it in the BSP, appropriately modifying its structure as explained below.

Algorithm details. Specifically, the main steps of the stochastic optimization process are the following (see Algorithm 1):

1. Initialize the root of the BSP tree with the entire parameter space (*low*,*high*) as bounds. Generate a uniformly sampled parameter value $\mathbf{x}$ as position. Find correspondences at $\mathbf{x}$ between point clouds using a GPU-accelerated method [Cay10]. Evaluate the cost function value according Equation (2), and store its value as current best. Define the maximum number of iterations i_M (we use $\tau = 1000$ as initial temperature).
2. Select a "promising" leaf according to the probabilistic scheme in Equation (3). The leaf is identified by taking left/right decision at each node.
3. Bisect the selected leaf and create new children, reassigning the old values to the child that contains the old sampling location; in contrast to previous work, we split nodes at longest edge rather than randomly choosing split planes. Bisection is performed so as to split the parameter space in portions of equal volumes [PB11].
4. Generate a new random sampling point $\mathbf{x}$ in the leaf that does not contain the old sampled value. Find correspondences at $\mathbf{x}$, evaluate the cost function value, and store its value as current best in the new node interval. Propagate bottom up the new parameter and function values so that each parent contains the best parameter and function value in both children.
5. Exploit the computed correspondences by finding a new parameter location $\mathbf{x}$ and function value $\varepsilon(\mathbf{x})$ through the solution of the absolute orientation problem in Equation (4).
6. Locate the leaf node containing the newly created position $\mathbf{x}$ through a top-down visit. Bisect the parameter interval in the leaf node into equal parts; if the new parameter value $\mathbf{x}$ and the one currently in the leaf are in different halves, create two children, one with $\mathbf{x}$, the other with the previous position; otherwise, if $\mathbf{x}$ improves the error value in the leaf, replace the values currently in the leaf node with the new ones. Finally, propagate bottom up.
7. Increment iteration count i_M by doubling the budget and repeat from step 2, otherwise terminate the algorithm if overlap between point sets is above a given threshold.

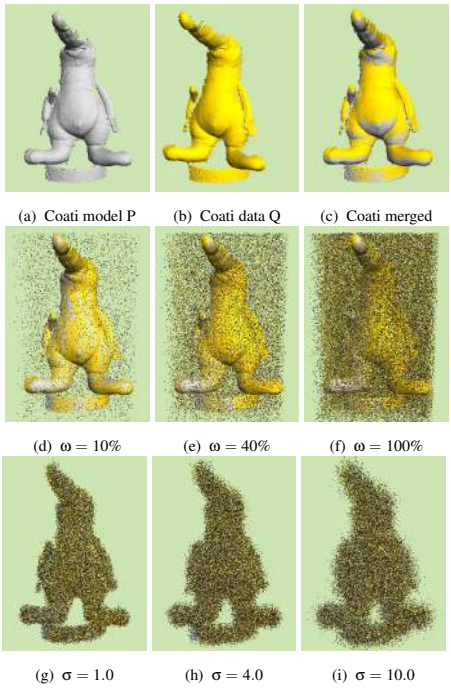
4. Results

We implemented our SOAR algorithm in C++ and tested it on a Linux system equipped with a 12 Intel Core i7-3930K 3.2 CPU Processor, 32GB RAM and a NVidia GeForce GTX 560.

Algorithm 1 Given two point sets P, Q in arbitrary positions compute the best affine alignment according to robust distance between the point sets and using stochastic branch and bound on a 7-dimensional parameter space in the interval $(low, high)$

INPUT: P, Q, $(low, high)$
OUTPUT: $\mathbf{x}_{opt} \in (low, high)$
if !tree **then**
  $\mathbf{x} \leftarrow rand(low, high)$
  compute matches between P and $T_{\mathbf{x}}(Q)$
  $f \leftarrow \varepsilon(\mathbf{x})$ using equation (2)
  $tree \leftarrow node(low, high, \mathbf{x}, f)$
end if
for $level = 1 \rightarrow L$ **do**
  $i_M = 2 * i_M$
  for $i = 1 \rightarrow i_M$ **do**
    compute p_l using equation (3)
    find $leaf = node(low_{leaf}, high_{leaf}, \mathbf{x}_{leaf}, f_{leaf})$ (select child with lower f if $rand(0,1) \leq p_l$)
    split in (low_{leaf}, mid) and $(mid, high_{leaf})$
    if $\mathbf{x}_{leaf} \in (low_{leaf}, mid)$ **then**
      $child_{left} \leftarrow node(low_{leaf}, mid, \mathbf{x}_{leaf}, f_{leaf})$
      $\mathbf{x}_{new} \leftarrow rand(mid, high_{leaf})$
      compute matches between P and $T_{\mathbf{x}_{new}}(Q)$
      $f_{new} \leftarrow \varepsilon(\mathbf{x}_{new})$ using equation (2)
      $child_{right} \leftarrow node(mid, high_{leaf}, \mathbf{x}_{new}, f_{new})$
    else
      $child_{right} \leftarrow node(mid, high_{leaf}, \mathbf{x}_{leaf}, f_{leaf})$
      $\mathbf{x}_{new} \leftarrow rand(low_{leaf}, mid)$
      compute matches between P and $T_{\mathbf{x}_{new}}(Q)$
      $f_{new} \leftarrow \varepsilon(\mathbf{x}_{new})$ using equation (2)
      $child_{left} \leftarrow node(low_{leaf}, mid, \mathbf{x}_{new}, f_{new})$
    end if
    propagate best to the root
    for $w = 1 \rightarrow W$ **do**
      try to improve $\mathbf{x}_I$,f_I by solving problem in eq. (4)
      find $leaf = node(low_{leaf}, high_{leaf}, \mathbf{x}_{leaf}, f_{leaf})$ containing $\mathbf{x}_I$
      split in (low_{leaf}, mid) and $(mid, high_{leaf})$
      if $\mathbf{x}_I \in (low_{leaf}, mid)$ & $\mathbf{x}_{leaf} \in (mid, high_{leaf})$ **then**
        $child_{left} \leftarrow node(low_{leaf}, mid, \mathbf{x}_I, f_I)$
        $child_{right} \leftarrow node(mid, high_{leaf}, \mathbf{x}_{leaf}, f_{leaf})$
      else
        if $\mathbf{x}_{leaf} \in (low_{leaf}, mid)$ & $\mathbf{x}_I \in (mid, high_{leaf})$ **then**
          $child_{left} \leftarrow node(low_{leaf}, mid, \mathbf{x}_{leaf}, f_{leaf})$
          $child_{right} \leftarrow node(mid, high_{leaf}, \mathbf{x}_I, f_I)$
        end if
      else
        if $f_I < f_{leaf}$ **then**
          $leaf \leftarrow node(low_{leaf}, high_{leaf}, \mathbf{x}_I, f_I)$
        end if
      end if
      propagate best to the root
    end for
  end for
end for
$\mathbf{x}_{opt} \leftarrow \mathbf{x}_{root}$

Approximate nearest neighbors are computed with the GPU-accelerated RBC library [Cay10]. Benchmarked KNN speeds were around 6M point queries per second (with $K = 4$ for fuzzy assignment), while error evaluation speed was 66M points queries per second.



(a) Coati model P (b) Coati data Q (c) Coati merged

(d) $\omega = 10\%$ (e) $\omega = 40\%$ (f) $\omega = 100\%$

(g) $\sigma = 1.0$ (h) $\sigma = 4.0$ (i) $\sigma = 10.0$

Figure 3: ***Robustness with respect to outliers and noise:*** *top row shows the reference model and data point sets for the Coati model together with the correct alignment, while middle and bottom rows show examples of alignments obtained with SOAR algorithm on both point sets degraded with varying amount of outliers ranging from $\omega = 10\%$ to $\omega = 100\%$ with respect to the original point count, and varying amount of gaussian noise with variance ranging from $\sigma = 1.0$ to $\sigma = 10.0$.*

Robustness with respect to outliers and noise. We evaluated the performance of our method regarding robustness with respect to noise and outliers. Specifically, we aligned two partially overlapping parts (containing 28.1K and 28.2K samples) of the Coati model (see Figure 3, top row), under varying conditions of noise and outliers added to both the model and the data set (see Figure 3 middle and bottom row). By adding considerable amounts of outliers, we also test for different amounts of overlap, in a range from 97% (original scans) to 48% (maximum amount of noise). We compared our results with the results obtained with the stochastic global registration method (in the following denoted as SGR) described in [PB11], which is currently assumed to be the best performing technique in the class of rigid stochastic registration methods, namely with respect to robust RANSAC schemes [AMCO08] and local descriptor based approaches [LG05]. Since the SGR method is limited to rigid registration, we kept in our method the scale parameter fixed to $s = 1$. For the sake of comparison, for each couple of datasets containing a different amount of outliers and noise we performed 20 trials with our SOAR method (4 multiple loops for a max of 15K iterations) and with SGR (one single loop with 15K iterations) and we recorded rigid transformation errors with respect to the correct solution. To improve accuracy, after stochastic optimization SGR was followed by 20 ICP steps. Charts in Figure 4 show the average alignment position errors together with error bars with respect to the correct ground truth transformation: SGR is compared to our method with different number of points used for fuzzy assignments ($K = 1$,$K = 4$, and $K = 8$). It is evident that our SOAR method is robust to outliers and noise and that, with same number of iterations, rigid alignment performances are generally better than those of SGR, especially when using $K \geq 4$ samples for fuzzy assignments.

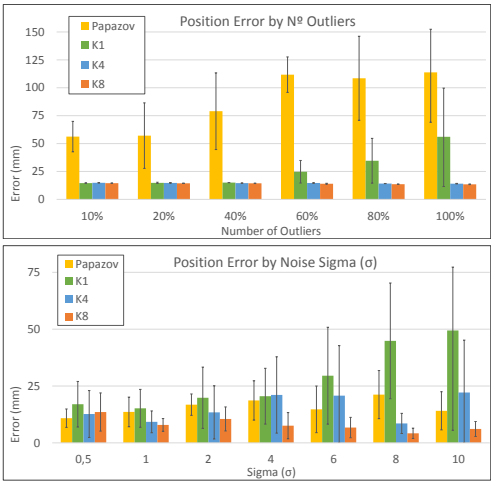


Figure 4: ***Statistical comparison of our method vs SGR with different amount of outliers and noise.*** *Position error in mm obtained for the registration of Coati (see Figure 3) with respect to ground truth are reported for SGR [PB11] (in yellow), and for SOAR with $K = 1$ (in green), with $K = 4$ (in blue), and with $K = 8$ (in red), showing its behavior in the presence of outliers (upper chart) and noise (lower chart).*

Affine alignment examples. An important use case of our method is cross-modality mapping. This problem occurs, for instance, when mapping sparse point clouds derived from structure-from-motion to denser clouds from range sensors [PGC11b, CDG*13] for the purpose of coloring datasets [PGC11a]. This problem leads to registration of sparse and noisy clouds at different scales, and cannot be handled with previous stochastic methods for rigid registration [AMCO08, PB11, MMA14]. Figure 5 shows successful alignments on very complex datasets (in red the sparse model point cloud registered with respect to the data point cloud).

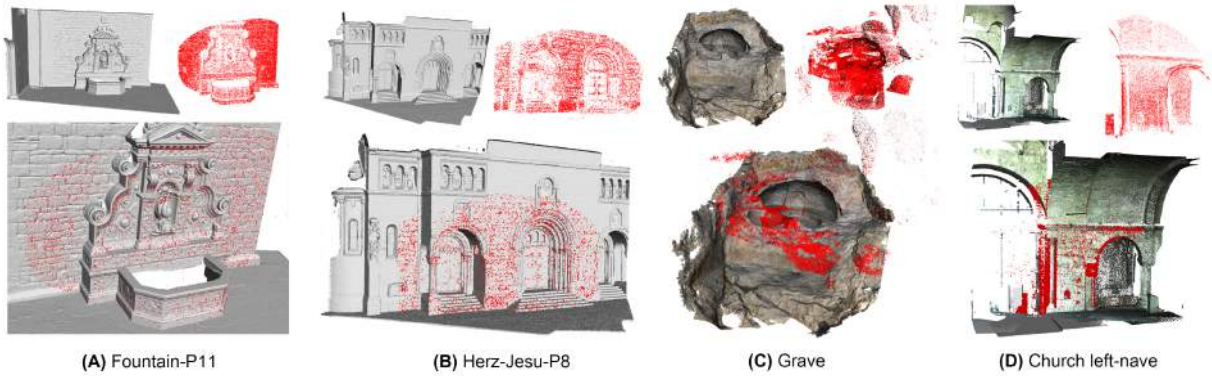


Figure 5: ***Multimodal affine alignment examples:*** *in red the data point clouds registered with respect to the model point clouds. Models come from Digital Heritage acquisition campaigns, and data are acquired with 3D laser scanners, and high resolution digital cameras.*

In all results, we used $K = 4$ for fuzzy assignments. Specifically, Figure 5 (A) and Figure 5 (B) show multimodal point clouds derived from models coming from the multi-view evaluation repository by Strecha et al. [SvHVG*08, SvHVG*]. The fixed model geometries are acquired with laser scanning (18Mpoints at 4mm resolution for Herz-Jesu and 13Mpoints at 3mm resolution for Fountain), while the data point clouds are derived from 6MP images acquired with digital cameras. Figure 5 (C) represents the digital model of an archaeological site (grave from a prehistoric necropolis). It was acquired using a time-of-flight laser scanner Leica ScanStation2 and a Nikon D200 camera; it is a challenging dataset since the geometry exhibits many smooth regions and does not contain well defined planes or straight lines. Finally, Figure 5 (D) represents a part of the left nave of a Romanesque Basilica, acquired using a time-of-flight laser scanner Leica ScanStation2. The data point clouds were instead derived from a photographic dataset acquired with a Nikon D200 camera. In this case, the automatic alignment is complicated by the fact that the model is non-uniformly sampled and contains a high percentage of outliers.

Dataset	Model	Data	Scale	Error (Overlap(%))
Fountain (Fig. 5 (A))	13M	58K	11.7	0.219(100)
Herz-Jesu (Fig. 5 (B))	18.1M	26.7K	0.85	0.09(100)
Grave (Fig. 5 (C))	8.4M	120.6K	280.0	10.984(91)
Church (Fig. 5 (D))	4.8M	63.5K	2093.0	0.201(94)

Table 1: ***Alignment statistics****: we report statistics on successful alignments obtained with our algorithm on various complex datasets exhibiting different scales and density. Datasets sizes, relative scales, as well as final scale normalized error values and overlaps are provided.*

Table 1 reports some statistics recorded during successful automatic alignments obtained with our algorithm.

Alignment times ranged from 3m22s to 47m22s for the most complicated case. Since for most datasets ground truth is not available, the quality of the final registration is verified by visual inspection (see Figure 5) and checked using additional knowledge about known correspondences (i.e., using the original images for the SfM datases). Figure 6 plots the value of the robust error function as function of the optimization time for the registration of datasets in Figure 5. As one can see, most alignments are obtained within few minutes. The dependence of optimization time on the number of correspondences employed for fuzzy assignment is presented in Figure 6(left). As one can note, optimization cost increases approximately linearly with the number of fuzzy correspondences. It should be noted, however, that this behavior is also dependent on the particular implementation of the KNN library employed (RBC [Cay10]).

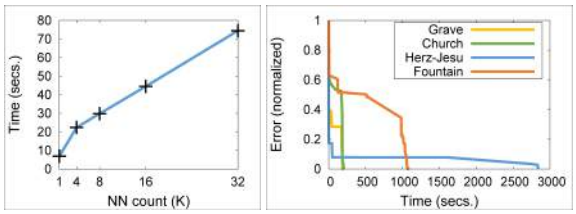


Figure 6: ***Performance statistics.*** *On the left, the registration time for Coati in Figure 3 top is reported as function of the number of nearest neighbors. On the right, the plot of normalized error function is reported as function of the optimization time for the registration of datasets in Figure 5.*

5. Conclusions

We presented an effective fully automatic method for solving the affine registration problem, which can also be used for the simpler rigid surface registration. The technique is based on a stochastic approach for the global optimization of an error function based on one-to-many distance metrics which are

robust to noise and resistant to outliers. The technique iterates between correspondence matching and error evaluation. In contrast to previous work, instead of simply constructing the tree by guided random sampling, we exploit the problem structure through a low-cost local minimization process based on analytically solving absolute orientation problems using the current correspondences. Our results demonstrate that the algorithm is capable to recover the globally optimal scale, rotation, and translation alignment parameters and is applicable in a variety of difficult settings, including very sparse, noisy, and outlier-ridden datasets that do not permit the computation of local descriptors. Current limitations of SOAR method are related to computational efficiency: as for other stochastic methods, long running times are sometimes required to reach the global optimum for complex models. To this end, we plan to exploit more efficient GPU parallel schemes to speed-up computations. Moreover, typical difficult cases for the method, shared with all other related methods working directly on points, occur when the objects exhibit strong slippage, leading to flat error landscapes. As future work, we plan to explore the potential of our approach for the reconstruction and processing of 3D architectural scenes starting from 2D images or videos, in order to build a fully automated 3D reconstruction pipeline.

Acknowledgments. This work was supported in parts by the People Programme (Marie Curie Actions) of the European Union's Seventh Framework Programme FP7/2007-2013/ under REA grant agreements n°290227 (DIVA).

References

[AMCO08] AIGER D., MITRA N. J., COHEN-OR D.: 4-points congruent sets for robust surface registration. *ACM TOG 27*, 3 (2008), #85, 1–10. 1, 2, 3, 6

[BS91] BILBRO G., SNYDER W.: Optimization of functions with many minima. *IEEE Trans. on Systems, Man and Cybernetics 21*, 4 (1991), 840–849. 3

[BTP13] BOUAZIZ S., TAGLIASACCHI A., PAULY M.: Sparse iterative closest point. In *Comp. Graph. Forum* (2013), vol. 32, Wiley Online Library, pp. 113–123. 2, 4

[Cay10] CAYTON L.: A nearest neighbor data structure for graphics hardware. In *First Int. Work. on Acceler. Data Management Systems Using Modern Processor and Storage Architectures* (2010). 3, 4, 5, 6, 7

[CDG*13] CORSINI M., DELLEPIANE M., GANOVELLI F., GHERARDI R., FUSIELLO A., SCOPIGNO R.: Fully automatic registration of image sets on approximate geometry. *Int. J. of Comp. Vision 102* (March 2013), 91–111. 1, 2, 6

[CSAD04] COHEN-STEINER D., ALLIEZ P., DESBRUN M.: Variational shape approximation. *ACM TOG 23*, 3 (Aug. 2004), 905–914. 2

[GMGP05] GELFAND N., MITRA N. J., GUIBAS L. J., POTTMANN H.: Robust global registration. In *Proc. Geometry Processing* (2005), pp. 197–206. 2

[GP02] GRANGER S., PENNEC X.: Multi-scale em-icp: A fast and robust approach for surface registration. In *Proc. ECCV* (2002), pp. 418–432. 2

[HB94] HECKER Y., BOLLE R.: On geometric hashing and the generalized hough transform, 1994. 2

[Hor87] HORN B. K. P.: Closed-form solution of absolute orientation using unit quaternions. *J. of the Opt. Soc. of America 4* (1987). 4

[HR09] HUBER P., RONCHETTI E.: *Robust Statistics*. Wiley, 2009. 3

[HW77] HOLLAND P. W., WELSCH R. E.: Robust regression using iteratively reweighted least-squares. *Comm. in Stat.-Theory and Methods 6*, 9 (1977), 813–827. 4

[JH99] JOHNSON A. E., HEBERT M.: Using spin images for efficient object recognition in cluttered 3d scenes. *IEEE Trans. Pattern Anal. Mach. Intell. 21*, 5 (1999), 433–449. 2

[JV05] JIAN B., VEMURI B. C.: A robust algorithm for point set registration using mixture of gaussians. In *Proc. ICCV* (2005), pp. 1246–1251. 2

[Kan90] KANATANI K.: *Group-theoretical methods in image understanding*, vol. 2. Springer-Verlag Berlin, 1990. 3

[LG05] LI X., GUSKOV I.: Multiscale features for approximate alignment of point-based surfaces. In *Symp. on geom. proc.* (2005), vol. 2, Citeseer. 6

[LH07] LI H., HARTLEY R.: The 3d-3d registration problem revisited. In *Proc. ICCV* (2007), pp. 1 –8. 1, 3

[MB03] M. T., BREUEL: Implementation techniques for geometric branch-and-bound matching methods. *Computer Vision and Image Understanding 90*, 3 (2003), 258 – 294. 1, 3

[MMA14] MELLADO N., MITRA N. J., AIGER D.: Super4pcs: Fast global pointcloud registration via smart indexing. In *Symp. on Geometry Processing* (2014), p. To appear. 2, 6

[OKO09] OLSSON C., KAHL F., OSKARSSON M.: Branch-and-bound methods for euclidean registration problems. *IEEE Trans. Pattern Anal. Mach. Intell. 31* (May 2009), 783–794. 3

[PB11] PAPAZOV C., BURSCHKA D.: Stochastic global optimization for robust point set registration. *Comput. Vis. Image Underst. 115* (2011), 1598–1609. 1, 2, 3, 4, 5, 6

[PG14] PINTUS R., GOBBETTI E.: A fast and robust framework for semi-automatic and automatic registration of photographs to 3d geometry. *ACM JOCCH* (2014). To appear. 3

[PGC11a] PINTUS R., GOBBETTI E., CALLIERI M.: Fast low-memory seamless photo blending on massive point clouds using a streaming framework. *ACM JOCCH 4*, 2 (2011), Article 6. 6

[PGC11b] PINTUS R., GOBBETTI E., COMBET R.: Fast and robust semi-automatic registration of photographs to 3d geometry. In *Proc. VAST* (October 2011), pp. 9–16. 1, 6

[Sto87] STOCKMAN G.: Object recognition and localization via pose clustering. *Comp. Vision, Graph., and Im. Proc. 40*, 3 (1987), 361 – 387. 2

[SvHVG*] STRECHA C., VON HANSEN W., VAN GOOL L., FUA P., THOENNESSEN U.: Dense multi-view stereo evaluation. http://cvlabwww.epfl.ch/ strecha/multiview/denseMVS.html. 7

[SvHVG*08] STRECHA C., VON HANSEN W., VAN GOOL L., FUA P., THOENNESSEN U.: On benchmarking camera calibration and multi-view stereo for high resolution imagery. In *Proc. CVPR* (2008), pp. 1–8. 7

[TARK10] TAMAKI T., ABE M., RAYTCHEV B., KANEDA K.: Softassign and em-icp on gpu. In *Proc. ICNC* (2010), pp. 179–183. 3

[TCL*13] TAM G., CHENG Z.-Q., LAI Y.-K., LANGBEIN F., LIU Y., MARSHALL D., MARTIN R., SUN X.-F., ROSIN P.: Registration of 3d point clouds and meshes: A survey from rigid to nonrigid. *IEEE TVCG 19*, 7 (July 2013), 1199–1217. 2

[TK04] TSIN Y., KANADE T.: A correlation-based approach to robust point set registration. In *Proc. ECCV* (2004), pp. 558–569. 2

[WR97] WOLFSON H. J., RIGOUTSOS I.: Geometric hashing: An overview. *Comp. in Science and Engineering 4* (1997), 10–21. 2