The 2009 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 11-15, 2009 St. Louis, USA

Pairwise Region-Based Scan Alignment

C. S. R. Aguiar, S. Druon, A. Crosnier
LIRMM, UMR 5506 Université Montpellier II - CNRS
161 rue Ada, 34000 Montpellier - France
aguiar,druon,crosnier@lirmm.fr

Abstract— **In this paper, we present a new algorithm for the alignment of two 3D scans. The approach uses a region-based matching technique. We make no assumptions about the initial positions of the scans. Regions are described by a probability density function (pdf) computed from low dimensional surface descriptors (curvature or normal cone). The algorithm allows registering directly raw noisy data, possibly with the presence of outliers, without any pre-processing, such as filtering, denoising, or reconstruction. Region correspondence is found using similarity function based on the comparison of regions pdf and under geometry constraints. Results on raw scan data sets are presented to illustrate and evaluate the algorithm.**

I. INTRODUCTION

The scans alignment, or global registration, problem consists in finding the common overlapping area between two 3D scans, called *model* and *data*, and in estimating a coarse rigid transformation that registers the *data* with respect to the *model*. This problem is encountered not only in the 3D registration pipeline but also in many other applications such as 3D shape retrieval, shape modeling, 3D object recognition, etc.

Commonly, global registration is done pairwise and it can be divided in two sub-problems. First, the portions of the two scans that overlap are established through point or region correspondence. Next, using a set of point correspondences on the overlapping area, the estimation of an initial transformation is performed. The result has to be sufficiently close to the correct registration pose in order to be used as the starting pose for automatic fine registration.

Missing data and holes currently occur in the input data due to occlusions and material reflections during the acquisition procedure. This raw data challenges the entire global registration methods in several ways. First, the input data set is too large for exhaustive point correspondence search. Second, noise and missing data affect the local geometry around a point and the global properties of the sample. Since the scans are in arbitrary initial position, the solution space of all possible alignment transformation is huge and search in this entire space is unfeasible.

Earlier work has identified several promising strategies that could be employed for pairwise scans alignment under these conditions. Nearly all global registration algorithms use a feature-based strategy. To deal with the growing data volume, salient features on both scans are identified and only a smaller sub-sample of *data* and *model* are used to match features under some rigidity (distance) constraint. The main drawback of feature-based approach is its performance directly related to the feature estimation.

Our global registration algorithm presents a different strategy to align pairs composed of two unstructured point clouds. We propose a region-based instead of feature-based approach. Region is preferred because of its descriptive power. Region representation also reduces the data volume and minimizes the noise effect on the global registration pipeline.

We use a local histogram as the region representation, which embeds the descriptor probability distribution over the region. The similarity function is based on histogram matching, and it gives a set of potentially corresponding region pairs. We assume that there is an one-to-one region correspondence. In addition, we increase the robustness of the matching by applying a combinatorial optimization to discard wrong correspondences, under the assumption that two point sets are related by a rigid transformation.

This article is organized as follows. In section II is presented some previous works on global registration. The input data and the local surface descriptor used are presented in section III. In section IV, an overview of our algorithm is presented, with the main steps of the pipeline. Region representation, matching algorithm and alignment transformation estimation is presented in section V. Experimental results on real scans are showed in section VI. Finally, we present the conclusions and the perspectives in section VII.

II. RELATED WORKS

Despite of the abundant literature about the problem of global registration, most commercially available systems still require user interaction for this step. The past research efforts to solve this problem concentrated mainly on matching local shape features [15]. Region-based matching for scans alignment has gained less attention, partly because of the lack of segmentation algorithms which produce repeatable regions [19]. While the first approach focuses mainly on selecting sparse local surfaces descriptors and comparing them over different scans, region approaches is less depend on the local feature itself and more dependent on the feature distribution over the region.

Feature-based approaches are categorized according to the dimension, or level of details, of their local surface descriptors. A survey on scan initial alignment based on features is presented in [15]. High-dimensional features used for global registration include spin-images [13], harmonic-shapes [20], point signatures, splash, etc. The advantage of high-dimensional descriptors is that, given a discriminant point in one scan, it is likely that it will be found only a

Fig. 1: Region-based global registration algorithm pipeline. This paper contributes the outlined stages of the global registration pipeline.

few points with similar descriptors. Scans alignment is solved through descriptors matching and outliers filtering. However, the scan alignment algorithms using high-dimensional descriptors are both time and memory consuming and they are difficult to compare efficiently.

Low dimensional descriptors, on the other hand, compute only a few values per point. Examples of such descriptors include curvature-based quantities [9], [18], [5], shape index [4], integral volume descriptor [7], [11]. These descriptors are computed around a small neighborhood. Consequently, many points on both *model* and *data* can have the same feature value. Since these descriptors are not discriminant, matching low-dimensional descriptors does not guarantee one-to-one robust feature correspondence. The scan alignment methods using these descriptors perform feature selection pre-processing, in order to find salient features and filter the set of points used in matching process.

Only a few region-based global registration algorithms were proposed in the literature [12]. Region-based global registration algorithms have more resemblance to the works on partial shape matching [6], [14], [17] and shape registration [12]. Compared to the matching problem, where the goal is to define a similarity function to compare shapes, the global registration problem have some additional issues, such as registration and integration.

In [12] is proposed a complete pipeline to solve the registration problem. They use a region-based global registration approach, where they partition the input dataset into homogeneous regions. They find the initial region correspondence set through region matching and use a forward search algorithm to find all correct surface matches. Our method differs from the one in [12] in both input data representation, region matching and correspondence strategy. First, while in [12] the input data is partition into homogeneous regions, we have regions with bounded descriptor variation. It provides regions with more dynamic and matching turns to be more stable, since regions are more discriminant. Both approaches use a set of corresponding regions to compute the initial alignment. However, while in [12] they aim at finding all corresponding regions and use a foward search algorithm for that, we find a minimum number of corresponding regions and use a branch-and-bound algorithm.

III. RAW POINT CLOUDS AND LOCAL SURFACE DESCRIPTORS

In our approach, we use raw point clouds as our modeling primitive. A 3D scan is then represented by an unstructured point cloud defined by $\mathbf{S} = (X, I)$. Let $X = \{\mathbf{x}_i\}$ be the finite set of sample points that approximates to some underlying piecewise smooth surface Σ, where the point $\mathbf{x}_i$ is defined by its 3D coordinates in the metric space $\mathbf{R}^3$. The mapping I assigns to each point $\mathbf{x}_i \in X$ a descriptor vector, $I(\mathbf{x}_i) \in \mathbb{R}^m$ which characterizes locally the surface around the point. This descriptor can be both estimated from the input data or provided by the acquisition system.

We are particularly interested in low-dimensional geometric descriptors, such as curvature and various curvature-based quantities [9], [18], [5], [7]. These descriptors measure how gently or strongly curved a surface is around a point. The properties that make low-dimensional descriptors appropriate to be applied to large data set are their invariability under rigid transformation and their low memory and computational cost. While the first property allows direct feature estimation and distribution comparison, the second yields into feature computation and comparison efficiency.

Any low-dimensional descriptors can be taken in our algorithm. In this paper, we use three possible geometric descriptors: the Mean and Gaussian curvatures [5], and the normal cone [2].

To compute these descriptors, the normal vector associated with the point must be estimated from the input data. We use a local neighborhood of a point which it assumed to be homeomorphic to a disc to estimate the normal vector. The neighborhood system is the sphere around a point $\mathbf{x}_i$, consisting of all points in X that are inside a sphere centered in $\mathbf{x}_i$ with a radius r, where r is proportional to the input data density [2]. The normal vector is estimated using the principal component analysis [10], minimizing in least square best fitting the plane formed by the neighborhood. Mean and Gaussian curvature is computed using the technique presented in [5]. In the next section we present an overview of the algorithm and its main steps.

IV. ALGORITHM OVERVIEW

We formulate our global registration algorithm as the problem of finding a set of region correspondences and use these correspondences to compute the initial alignment transformation. The algorithm pipeline is illustrated in figure 1.

Let $\mathbf{S}_A$ and $\mathbf{S}_B$ be the *data* and the *model* raw point clouds, respectively. After segmentation, both *data* and *model* clouds are partitioned into a set of regions, $\mathcal{P}_{\mathbf{S}_A} = \{A_1, \ldots, A_n\}$ and $\mathcal{P}_{\mathbf{S}_B} = \{B_1, \ldots, B_m\}$. Segmentation step aims at reducing the volume of data treated without loosing

discriminant information about initial scans. A region is characterized by a descriptor vector. This descriptor vector is composed by the region barycenter and an histogram representing the probability density function of a geometric descriptor over the region.

In the case of rigid motion, three corresponding points are sufficient to uniquely determine the alignment transform. Working in the region space, the alignment transform is determined by three corresponding regions. Assuming that, on the overlapping area, there are at least three corresponding region pairs, a naive alignment scheme has a time complexity of $O(m^3n^3)$: for each region triplet from $\mathcal{P}_{\mathbf{S}_A}$ take a set of three regions from $\mathcal{P}_{\mathbf{S}_B}$, solve the unique rigid transformation using this correspondence, and evaluate the quality of the current transform. This solution requires to compute the transformation for the entire set [7] and it becomes overly expensive as the number of regions grows. In the present paper we propose an algorithm to explore efficiently the search space of possible set of corresponding region triplet pairs.

The correspondence between two regions A_i and B_j is establish through their equivalence relation. We say two regions A_i and B_j are equivalent if it is observed similarity in their absolute region properties and geometric consistency, or relational constraints, among regions on the overlapping area.

We use the first equivalence condition to build an initial set of potential correspondences. This set is built by finding, for each region $A_i \in \mathcal{P}_{\mathbf{S}_A}$ in the *data*, a set of regions $C(A_i) = \{B_j\} \subset \mathcal{P}_{\mathbf{S}_B}$ in the *model* which have similar attribute probability distribution. Intuitively, two regions A_i and B_j potentially correspond if they have similar attribute distribution.

The second equivalence condition states that the geometric constraints given by the transformation class that is considered for registration (e.g. rigid objects, rigid transformation, etc.) result in a set of conditions that need to be verified between correspondences. From body rigidity hypothesis, we have that any rigid transformation has to preserve inter-point distance, which is based on the distance between two points on the same point cloud. We use the internal pairwise distance to build, from the initial potential correspondence set, a set of potential region triplet pairs. In this way, we avoid the computation and evaluation of the alignment for each potential pair of region triplets. Furthermore, this step filters wrong correspondences and, consequently, reduces the search space of possible correspondences.

Finally, we have only a small set of region triplet pairs that potentially are in correspondence. For each region triplet pair in this set, it is computed the candidate transformation that aligns the region triplet pairs. The initial alignment transformation is the one that minimizes the distance between the two point sets. In figure 2 we shows the main steps of the pipeline and the resulting alignment on the Bunny dataset.

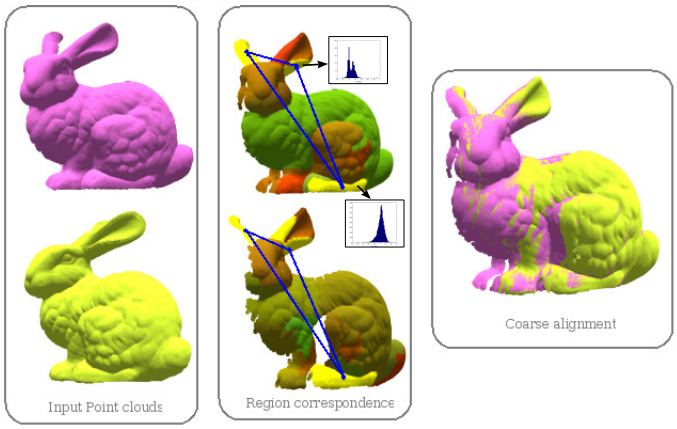


Fig. 2: Region-based global registration algorithm. First column: raw point clouds. *Data* (top) and *model* (bottom). Second column: regions obtained from segmentation and region triplet pairs that produces the best initial alignment. Third column: point clouds after alignment transformation is applied.

V. PAIRWISE REGION-BASED MATCHING

As explained earlier, the algorithm relies on triplets of regions to estimate the transformation. We developed a branch-and-bound algorithm to search efficiently the solution space of possible regions correspondences. Finding the best triplet of corresponding regions is done by descending the decision tree, incrementally making choices about which pairs of regions should be in the correspondence. This is a greedy algorithm, since it never backtracks to reconsider the choice.

A. Region Representation

In order to find region correspondences, we derive a concise representation of each region. We store the following information for each region R_i:

$$R_i \longrightarrow \{\mathbf{c}(R_i), |R_i|, H(R_i)\} \tag{1}$$

where,

- The point $\mathbf{c}(R_i)$ represents the *position* of the region, defined as the closest point to the barycenter of the region.
- $|R_i|$ is the size (number of points) of the region R_i.
- $H(R_i)$ contains the signature of the region R_i. The mapping I defines the geometric descriptor and associates with each point $\mathbf{x} \in R_i$ a value $I(\mathbf{x})$. In this paper, we assume that $I(\mathbf{x}) \in \mathbb{R}$ is one dimension. $H(R_i)$ is the descriptor values histogram of points $\mathbf{x} \in R_i$.

The histogram $H(R_i)$ is a non-parametric representation of the probability density function of a region. Histograms have been used in both 3D global shape representation [3] and high-dimensional local descriptors [13], [20]. We preferred a non-parametric pdf representation over a parametric

one because of its ability to discriminate higher dimensional information.

To be able to compare histograms, we will construct a histogram for every region pair A_i and B_j with the same parameters. The number of bins β in the histogram is computed using Scott's rule [16], $\beta = 3.49\sigma_I N^{-\frac{1}{3}}$, where σ_I is the standard deviation of the N points. We take $N = |A_i|+|B_j|$ and σ_I is computed over the data represented by A_i and B_j.

Then, each histogram $H(A_i)$, $H(B_j)$ is characterized by:

$$H_{\min} = \min\{I(\mathbf{x})_{\mathbf{x}\in A_i}, I(\mathbf{x})_{\mathbf{x}\in B_j}\} \quad (2)$$

$$H_{\max} = \max\{I(\mathbf{x})_{\mathbf{x}\in A_i}, I(\mathbf{x})_{\mathbf{x}\in B_j}\} \quad (3)$$

Regions are obtained through the segmentation of the input datasets. We formulate 3D point cloud segmentation as a graph partition. The point cloud is partitionned into subsets (regions), where the maximum descriptor variation between region points is bounded. For this, we use a modified version of the data-driven MST-based segmentation algorithm presented in [1], [2], where the partition criteria relies on region homogeneity.

Let $\mathbf{S}$ be the input point cloud. The MST-based segmentation algorithm partitions the input into n subsets or regions, $\mathcal{P}_\mathbf{S} = \{R_1, \ldots, R_n\}$. Each region $R_i \in \mathcal{P}_\mathbf{S}$ is represented by a set of points that are linked by a unique tree with minimal cost. The resulting edges have weights that are ε-bounded in descriptor space. In the original algorithm, ε is a fixed input parameter. It bounds the tolerated noise level. Here, we estimate ε from the input data. Given the point cloud $\mathbf{S}$, we compute a histogram of the descriptor variation on the edges. This histogram is an approximation of the probability density function (pdf) that represents the variation of the descriptor over the region. We choose ε as the value on the pdf function that guarantees to cover $K\%$ of the input edge set.

Data-driven segmentation algorithms sometimes result in an over-segmentation or under-segmentation of the same object and they can produce different results under slightly different acquisition conditions [19]. The main advantages of using the MST-based segmentation algorithm over other data-driven methods are that the solution space of all possible partitions is drastically reduced and that the noise can be filtered during segmentation process [2].

B. Pairwise region correspondence

There is a large number of dissimilarity functions to compare histograms, such as euclidean distance, quadratic form distance [3], statistical and probabilistic approaches [13], [8], among others. In this paper we restrict ourselves to statistical similarity functions.

The formal statistical method for assessing the dissimilarity between two probability functions is the χ^2-test [8]. Another equivalent comparison measure to histograms is the intersection measurement, a similarity function which quantifies the common parts of two histograms. The intersection of two histograms $H(A_i)$ and $H(B_j)$ is defined as:

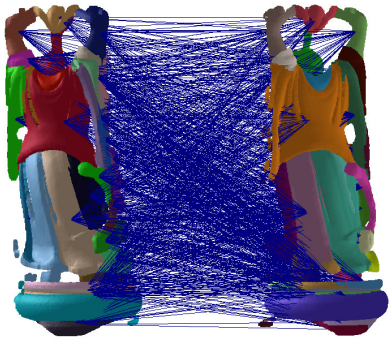


Fig. 3: The initial of potential correspondences for the Happy Buddha dataset.

$$\cap(H(A_i), H(B_j)) = \frac{1}{\beta}\sum_k \frac{\min(a_k, b_k)}{\max(a_k, b_k)} \quad (4)$$

where β is the number of bins on the histograms and a_k, b_k are the k-th histogram cell. Equation (4) is the normalized intersection function presented in [8] and it assumes values in the range $[0, 1]$. We define the dissimilarity cost function between two regions A_i and B_j from the intersection similarity as :

$$Ds(A_i, B_j) = [1 - \cap(H(A_i), H(B_j))]^2 \quad (5)$$

The dissimilarity function $Ds(A_i, B_j)$ assumes values in the range $[0, 1]$. The main advantage of this measurement is that it does not consider regions shape or points location, which is an advantage when matching partial overlapping regions. Furthermore, all histogram cells are supposed to be equally probable. Additionally, this distance function defavours regions of similar sizes.

The initial set of potential correspondences for each region $C(A_i)$ is found by taking all regions $B_j \in \mathcal{P}_{\mathbf{S}_B}$ where the dissimilarity between the regions histograms is such that $Ds(A_i, B_j) < \varepsilon_{Ds}$, where ε_{Ds} is an user-defined parameter. Figure 3 illustrates the initial correspondence set of the Happy Buddha dataset.

C. Finding triplets correspondence through geometric consistency

Given a initial set of potential correspondences $\{C(A_0), C(A_1), \cdots, C(A_n)\}$, the goal of the correspondence search algorithm is to find three corresponding regions to estimate the alignment transform. One way to evaluate the quality of the correspondence is to compute the alignment transformation for each possible triplet, or the cRMS (*coordinate root mean squared error*) distance, which has been showed to be overly expensive (see section 2). Another solution is a pairwise comparison of internal distance between regions taken in each set. Under rigid transformation assumption, we have that for any region pair (A_i, B_i^l) and (A_j, B_j^k) in correspondence, the distance between the pairs $(\mathbf{c}(A_i), \mathbf{c}(A_j))$ should be the same as

between their corresponding regions $(\mathbf{c}(B_i^l), \mathbf{c}(B_j^k))$. The error metric based on inter-point distance is known as *distance root mean squared error*, or dRMS.

We will use the last solution, once it has the advantage to avoid the computation of the transformation that aligns the two point clouds. The *distance root mean square error* dRMS is defined as

$$dRMS^2 = \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{n}(\|\mathbf{c}(A_i)-\mathbf{c}(A_j)\|-\|\mathbf{c}(B_i^l)-\mathbf{c}(B_j^k)\|)^2 \tag{6}$$

where n is the number of correspondences for the given regions $B_i^l \in C(A_i)$ and $B_j^k \in C(A_j)$. The dRMS function is used as an alternative distance function in [7] to robustly find a set of corresponding features. They proved the equivalence between the dRMS and the cRMS functions, by showing that dRMS is both lower and upper bounded by cRMS.

The main advantage of using dRMS to evaluate potential correspondences is that it does not require to compute the alignment transformation. In fact, dRMS cost is only paid once for every point pair, since it compares intrinsic properties of two sets of corresponding points, namely the internal pairwise distances of each pointset. It means that, for each point cloud, the internal pairwise distance of its features is computed only once and stored in a matrix. To compare potential correspondence sets in a second point cloud, it is sufficient to compare their respective matrices.

D. Algorithm

The input of the algorithm is a region set taken from both *data* and *model*. Each region is represented by the descriptor vector (1). The correspondence problem is viewed as a search over all possible sets of regions in the *model*, where we want to find three corresponding region pairs that minimizes the distances functions (5) and (6). Our global registration algorithm proceeds as follow:

1) **Build the initial potential correspondence set:** For each region A_i, we will designate the j-th member of the potential correspondence set $C(A_i)$ as B_i^j, if $Ds(A_i, B_j) < \varepsilon_{Ds}$.
2) **Form region pairs:** For each region pair $(A_i, A_j) \in \mathcal{P}_{S_A}$, it is taken the set of corresponding pairs $(B_i^o, B_j^p) \in \mathcal{P}_{S_B}$, where $B_i^o \in C(A_i)$ and $B_j^p \in C(A_j)$, characterized by a dRMS (6) less equal than a given threshold ε_{dRMS}. It leads to a set of potential initial correspondences. We sort this set in order of increasing distance discrepancy.
3) **Add a region to form triplets:** From the set of potential corresponding region pairs, we traverse the search space looking for a third correspondence, (A_k, B_k^q). If we do not find a third potential correspondence, we remove the correspondence pair from the potential pair. Otherwise, we find the region pair (A_k, B_k^q) that minimizes the dRMS function.
4) **Prune region triplets:** At the end of steps 1 and 2, we obtain a set of potential triplets characterized both by a minimum dissimilarity (5) and a minimum $dRMS$ error (6). A prune process is performed in order to retain triplets that minimize both distance functions. We sort this set in order of increasing $dRMS$ distance discrepancy, followed by a pruning. We repeat the same procedure using the region dissimilarity cost.
5) **Registration test:** Once only a few potential triplets are left, we apply the coarse alignment transformation and take the triplet corresponding to the regions with the minimum $cRMS$ cost.

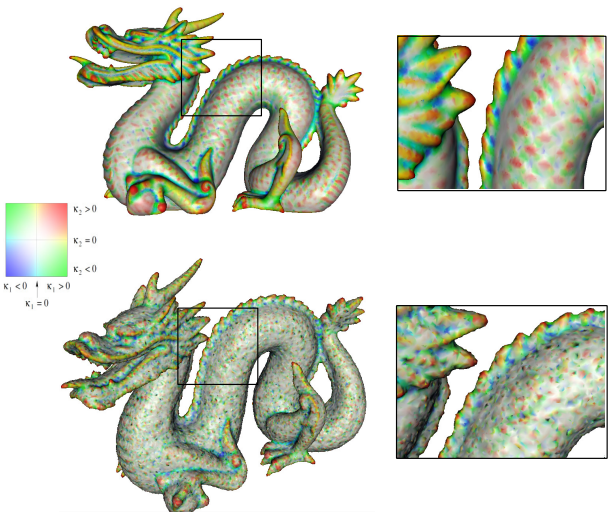
The basic idea is, at each step of the algorithm, to narrow the solution space of potential correspondences and increase the complexity of the operations performed. At each step, it is removed correspondences that do not respect one of the equivalence conditions. The algorithm is efficient because it only computes the alignment transformation and the alignment error (the most expensive operation of the pipeline) for a small subset of the initial search space.

VI. EXPERIMENTAL RESULTS

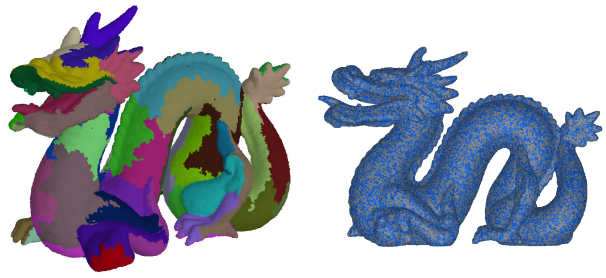
We tested our region-based global registration algorithm on a variety of input data with varying amount of noise, outliers, and extent of overlap. We now report performance regarding the robustness to noise and to partial region overlapping. Figure 2 shows the main steps of the algorithm and the resulting transform when our algorithm is applied on the Stanford Bunny scans dataset. Although in the examples the *model* and *data* point clouds are shown in similar positions, we stress out that algorithm does not depend on any assumptions about the initial positions of the input point clouds.

Figure 4 shows the robustness of the region-based global registration algorithm under a zero-mean Gaussian noise without any ICP refinement. We align the Stanford Dragon model to a copy of itself corrupted by zero-mean Gaussian noise. Figure 4 (a) shows the *principal curvature* colormap, and illustrates how noise affects the estimation of geometric descriptors. We set the segmentation partition threshold $K = 30\%$ for both *data* and *model*. Figure 4 (b) shows the resulting regions after segmentation. Despite the noise affects geometric descriptor values, the regions of both *data* and *model* were exactly the same. It shows the repeatability of the segmentation [1] under noise. In this scenario, we have that each region in both point clouds has the same barycenter, but different histograms. The initial potential correspondence set was built using $\varepsilon_{Ds} = 0.5$. Our alignment brings the *data* (noisy) point cloud into exact alignment to the *model* (smooth) point cloud (fig. 4 (c)).

Figure 5 shows the robustness of the region-based global registration algorithm under different segmentations. We align two raw point clouds of the Stanford Happy Buddha model. The magnitude of the segmentation partition threshold ϵ is varied in a scale where $K = 30\%$ was taken as one unit. Figure 3 shows the initial potential correspondence set for $K = 30\%$ .The pose computed by our algorithm is refined by running two iterations of ICP and the alignment error is computed using the cRMS point-point error metric [21].

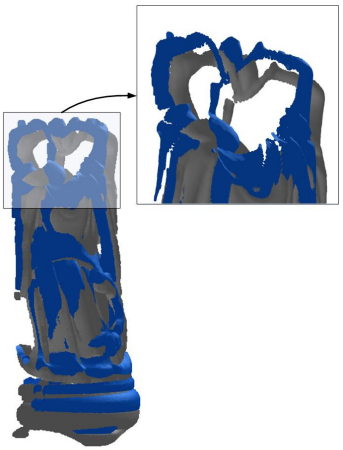


(a)

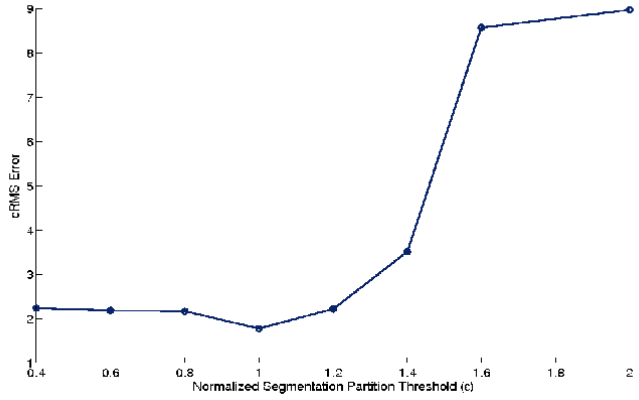


(b) (c)

Fig. 4: Dragon example. (a) Input to the region-based global registration algorithm: raw point cloud (the *model*) and the noisy dragon (the *data*) with the curvature colormap. (b) Regions obtained after segmentation. (c) Registration after applying our algorithm.



(a)



(b)

Fig. 5: Evaluation of the initial alignment quality under segmentation variation. (a) Initial alignment when $\epsilon = 1$. Detail shows the error in the alignment. (b) Graph of cRMS error as the function of the segmentation partition threshold ϵ.

The results are showed in figure 5(b). For all segmentations, our region-based global registration converges to a correct alignment.

When $K < 30\%$, the point clouds are over-segmented. Thus, the number of regions per point cloud is larger, and, consequently, the search space of possible region correspondences. The alignment error for over-segmented regions is similar to the error of a correct segmentation. The algorithm robustness to over-segmentation is due to the three region correspondence hypothesis. As the regions get small, region histogram descriptor is not enough discriminant and the alignment relies mostly on the geometric constraint. Notice how the alignment error increases when the input point clouds are under-segmented. This is because there are less regions and a smaller search space of possible correspondences. In this condition, the algorithm only finds the correct alignment because both point clouds have a large overlapping area.

VII. CONCLUSION

In this paper, we have presented an algorithm that solves the alignment problem using regions correspondence for two three-dimensional scans without any assumption about their initial position. The main contribution of the work concerns the use of a region-based matching to estimate the coarse alignment transformation. We proposed an algorithm to search efficiently the solution space of possible regions correspondences. Our algorithm is able to align whole and partially overlapping shapes, and is robust to noisy data and unstable segmentation. Experimental results on raw point clouds have showed that the algorithm finds the coarse alignment even under noise and partial overlapping. In the future, we would like to study the exact relationship between the size of the regions, the geometric descriptors and the performance of global registration to develop a method to estimate robustly the input parameters.

VIII. ACKNOWLEDGMENTS

This research is supported by French National Research Agency (Agence Nationale de la Recherche). Point set models were provided courtesy of Stanford Scanning Repository, Digital Michelangelo Project (Stanford).

REFERENCES

[1] C. S. R. Aguiar, S. Druon, and A. Crosnier. 3D datasets segmentation based on local attribute variation. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2007)*, pages 571–578, San Diego, CA, USA, 2007. IEEE Computer Society.

[2] C. S. R. Aguiar, S. Druon, and A. Crosnier. Hierarchical segmentation for unstructured and unfiltered range images. In *4th International Conference on Computer Graphics, Imaging and Visualization (CGIV 2007)*, pages 261–267, 2007.

[3] M. Ankerst, G. Kastenmaller, H.-P. Kriegel, and T. Seidl. 3D shape histograms for similarity search and classification in spatial databases. In *Advances in Spatial Databases, 6th International Symposium, SSD'99*.

[4] C. Dorai and A. K. Jain. COSMOS - a representation scheme for 3d free-form objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(10):1115–1130, 1997.

[5] P. Flynn and A. Jain. On reliable curvature estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '89)*, pages 110–116, Washington, DC, USA, 1989. IEEE Computer Society.

[6] R. Gal and D. Cohen-Or. Salient geometric features for partial shape matching and similarity. *ACM Trans. Graph.*, 25(1):130–150, 2006.

[7] N. Gelfand, N. J. Mitra, L. J. Guibas, and H. Pottmann. Robust global registration. In *Proc. Symp. Geom. Processing*, pages 197–206, 2005.

[8] G. Hetzel, B. Leibe, P. Levi, and B. Schiele. 3D object recognition from range images using local feature histograms. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '01)*, 2:394, 2001.

[9] A. Hoover, G. Jean-Baptiste, X. Jiang, P. J. Flynn, H. Bunke, D. B. Goldgof, K. Bowyer, D. W. Eggert, A. Fitzgibbon, and R. B. Fisher. An experimental comparison of range image segmentation algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(7):673–689, 1996.

[10] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle. Surface reconstruction from unorganized points. *Computer Graphics*, 26(2):71–78, 1992.

[11] Q.-X. Huang, S. Flöry, N. Gelfand, M. Hofer, and H. Pottmann. Reassembling fractured objects by geometric matching. *ACM Trans. Graphics*, 25(3):569–578, 2006.

[12] Q.-X. Huang and H. Pottmann. Automatic and robust multi-view registration. Technical Report 152, Geometry Preprint Series, Vienna Univ. of Technology, December 2005.

[13] A. Johnson and M. Hebert. Surface registration by matching oriented points. In *International Conference on Recent Advances in 3-D Digital Imaging and Modeling*, pages 121–128, May 1997.

[14] R. Osada, T. Funkhouser, B. Chazelle, and D. Dobkin. Matching 3D models with shape distributions. In *SMI '01: Proceedings of the International Conference on Shape Modeling & Applications*, page 154, Washington, DC, USA, 2001. IEEE Computer Society.

[15] B. M. Planitz, A. J. Maeder, and J. A. Williams. The correspondence framework for 3D surface matching algorithms. *Comput. Vis. Image Underst.*, 97(3):347–383, 2005.

[16] D. W. Scott. On optimal and data-based histograms. *Biometrika.*, 66(3):605–610, 1979.

[17] J. W. H. Tangelder and R. C. Veltkamp. A survey of content based 3D shape retrieval methods. In *Shape Modeling Applications, 2004. Proceedings*, pages 145–156, 2004.

[18] R. Unnikrishnan, J.-F. Lalonde, N. Vandapel, and M. Hebert. Scale selection for the analysis of point-sampled curves. In *Third International Symposium on 3D Processing, Visualization and Transmission (3DPVT 2006)*, June 2006.

[19] N. A. Varsha Hedau, Himanshu Arora. Matching Images Under Unstable Segmentation. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '08)*, 2:394, 2008.

[20] D. Zhang, K. Ikeuchi, and D. Zhang. Harmonic shape images: A 3-D free-form surface representation and its application in surface matching. In *Ph.D. dissertation, Carnegie Mellon Univ*, 1999.

[21] Z. Zhang. Interative point matching for registration of free-form curves and surfaces. *Int. Journal of Computer Vision*, 13(2):119–152, 1994.