HSPA: Hough Space Pattern Analysis as an Answer to Local Description Ambiguities for 3D Pose Estimation

Fabrice Mayran de Chamisso
fabrice.mayran-de-chamisso@cea.fr
Boris Meden
boris.meden@cea.fr
Mohamed Tamaazousti
mohamed.tamaazousti@cea.fr

Université Paris-Saclay, CEA, List,
F-91120, Palaiseau, France

Abstract

When performing feature-based 3D object registration, one may expect to find a unique point corresponding to the right transformation in Hough space for each object instance. However, we observed that description ambiguities of the objects or scenes create a structured pattern in the Hough space of transformations during the matching process. We argue that this pattern can be viewed as a global descriptor, as opposed to the local descriptors or features whose matching resulted in the pattern. Thus, we propose to shift the focus from finding better local descriptors to better using the Hough-space pattern. This paper introduces a methodology to compute, analyze and match said patterns in order to improve the quality of 3D pose estimation. We detail a whole framework, termed HSPA, to first generate what we call the Hough space canonical invariance pattern for any given object to register and second, take this pattern into account when assembling and pruning pose hypotheses generated by a registration algorithm. We show the benefits of this technique on object registration as well as 3D scene registration benchmarks.

1 Introduction

Rigid object localization aims at finding a six degree of freedom (6DoF) transformation from a rigid object in the scene to the same object in a reference configuration. Typically, the scene is a point cloud, RGB or RGB-D image captured by a sensor and the reference is a CAD model or a point cloud obtained by fusing multiple sensor acquisitions with varying point of view. There are a large number of localization approaches: template-based approaches such as [20, 24], whole-image based processing using neural networks such as [40] or [36] and many descriptor-based approaches, reviewed in [16]. However, all these approaches eventually converge to a set of pose hypotheses: 6DoF transformations with a confidence score. The space of the degrees of freedom of transformations is called *generalized Hough space*. In the ideal case, for objects without any symmetry, there is a unique rigid transformation from any given scene to a given reference, resulting in theory in a single point in Hough space corresponding to the object's localization.

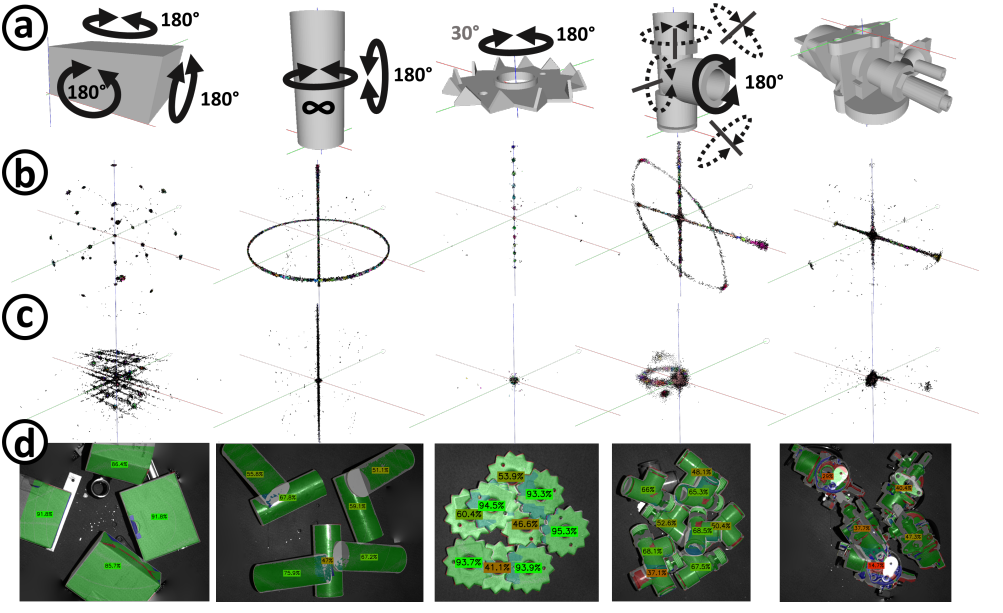

Figure 1: Canonical invariance patterns and scene-to-model registrations for objects of the ITODD [11] dataset. a) objects with invariance and almost-invariance axes. b) rotation and c) translation parts of the Hough space. d) registrations obtained by the proposed method, with confidence scores from red to green. Even when an object has no apparent symmetry (pump object, right), some parts may still exhibit invariances.

In practice, transformation hypotheses are noisy, resulting in multiple points in Hough space forming a cluster around the correct transformation. This is accounted for in Hough-space pose grouping approaches [15, 16, 27, 28, 29, 30, 42, 47]. What is not accounted for, however, is the fact that objects with global symmetries such as planes, cylinders, spheres or gears result in multiple and possibly disjoint clusters in Hough space. More generally, when objects have parts that resemble each other relatively to the descriptor used[1] (or feature vector, template, etc.), which we call *invariances*, structured patterns can be observed in Hough space (see Figure 1). We call these *invariance patterns* because they arise due to the description of the object being locally or globally invariant. In other words, there exist transformations, which, when applied to the object, leave a subset of its descriptors invariant. Invariance patterns are implicitly considered detrimental to localization in state of the art approaches. On the contrary, in this paper, we develop a methodology to exploit them. Our contributions are as follows: (i) we build a global Hough-space descriptor called the canonical invariance pattern. (ii) We propose a Hough-space pattern matching algorithm for 3D pose estimation. (iii) We propose analgorithm to find and exploit symmetry-breaking details that we call “disambiguation”. (iv) We propose a local descriptor to supply Hough-space pose hypotheses with competitive computation and matching times.

[1]The limit between “same” and “different” descriptors or "symmetrical" and "non-symmetrical" objects is often a question of level of detail of the description.

2 Related Work

Symmetries and Hough space in 3D geometry Mitra, Pauly et al. [32, 33, 37] detect pairs of points symmetric along a line in Hough space and perform clustering in order to find global symmetries. This allows them to symmetrize real-world objects (see Figure 15 of [33]) as well as, for instance, edit geometry while preserving a symmetry property. Also, by matching two point clouds representing the same object during a motion, they obtain one cluster per rigid body which allows easy segmentation thereof. All analyses are based on clustering sets of Hough points obtained by matching invariant zones. One of the key differences with our work is that Mitra et al. do not consider votes for the pose of an object, but for a symmetry axis, which prevents the use of an invariance pattern for 3D pose estimation.

Symmetries and Hough space for 3D pose estimation The methods addressing symmetry issues for 3D pose estimation can be categorized in two families. The first category assumes information on the set of object symmetries to improve scoring of poses. [8, 9] remove poses that are symmetries of each other from Hough space. A similar strategy is used in [36]. [1] uses symmetries to reduce the Hough space used for pose clustering by only keeping one instance of each symmetrical part of the original object. Finally, [39] introduces a pipeline to handle symmetrical or quasi-symmetrical objects whose aspect is the same from multiple points of view. By modifying the loss function driving a neural network to predict symmetry-invariant data, the average recall in an object localization task is improved and the learning time is reduced. Global symmetries are handled in a binary way (symmetrical/not symmetrical pose), ignoring objects with local symmetries (only symmetrical parts). Fine localization of symmetry-breaking details (foolproofing, holes, ...) is not handled in [39].

The second category is approaches *implicitly* robust to symmetries to some extent. For instance, [22] splits objects into fragments. Fragment matches and object labels are estimated separately, so that a single object label can be predicted even in the case of symmetries. The symmetries themselves are not considered during inference. During learning, symmetries are not taken into account either, with the hypothesis that, over the whole dataset, all symmetries will be represented. It is not clear how this approach is impacted by local symmetries in an object. However, this method uses the implicit assumption that some of the fragments are symmetry-disambiguating, so that matching them will result in an ambiguous pose. In the same category, [28] finds poses representing the same model but with a different symmetry by using an ICP score (average of closest-point distances) to compare poses, based on the fact that the geometry is invariant by any of its symmetries. This process is computationally expensive (at least $n \log n$ per pose pair). However, one can naturally expect the ICP score to handle local invariances to some extent.

Note that none of these approaches provides a way to *find the set of symmetries*. In this paper, we propose to build this set of symmetries in the form of our Invariance Hough Space Pattern as shown in Figure 1.

Feature-based 3D pose estimation In order to better understand our method, we give a quick overview of feature-based 3D pose estimation. One way to tackle the problem is end-to-end learning [2, 26] or [13] with RGB-D input. However, to the best of our knowledge, there is no trivial way to tackle the symmetry issues except by representing all instances of the symmetries in the input dataset. The problem seems to be ill-posed as noted in [39]. For this reason, we focus on feature-based approaches which can be summarized in three

classical steps: 1) data preparation (smoothing, normal and sometimes reference frames calculations), keypoint detection, etc., 2) description and matching, and finally 3) outlier rejection and pose refinement for robust alignment.

Regarding description, there are traditional/geometrical and learning-based methods. There are currently two major geometry-based 3D registration pipelines. The first one is based on descriptors such as SHOT [42] computed on a 3D point cloud in a local reference frame centered on each keypoint. Each descriptor match generates a unique 6D pose hypothesis. The second pipeline is based on Point-Pair Features (PPF), as exposed notably in [12]. PPF do not use reference frames but require both points of each pair to match (2 to 2 matching), which implies a quadratic sampling of points. The quadratic computation cost is avoided by smart sampling of computation points [21]. Otherwise, many approaches have taken advantage of Deep Neural Networks for keypoint detection [3, 4] and feature description [5, 25]. All in all, state of the art approaches focus on having each pose hypothesis correct up to noise, which leads to developing more and more complex descriptors (for instance SHOT [43] with 352 dimensions), templates or neural networks to generate hypotheses. Mismatches will happen even with very complex descriptors, since there will always be objects or parts of objects invariant under description by the chosen descriptor. Of particular interest is the case of symmetrical objects such as cylinders, for which the notion of a "correct" pose is undefined and the symmetry will manifest in any description of the object.

Regarding outlier rejection, existing methods are either RANSAC [22, 35] or Hough based [12, 16, 17, 27, 28, 29, 30, 31, 42, 47]. Hough based approaches mainly differ by the number of dimensions considered: 3 (translations or rotations only), 6 (both) or 7 (with a scale factor) and by the clustering method (voxel-based or sparse). A comprehensive review of traditional 3D outlier rejection is proposed by [45]. Recently, some deep-learning based methods for outlier rejection have been proposed [6, 10].

3 Our Hough Space Pattern Analysis (HSPA) approach

3.1 Standard 3D registration pipeline

In order to perform invariance analysis in Hough space, we first have to populate said space with pose hypotheses. In this paper, we are not interested in how pose hypotheses are generated, so we choose a classical descriptor-based pipeline to simplify explanations. Such pipelines are reviewed in [16]. The only requirement in order to perform invariance analysis in Hough space is to generate multiple pose hypotheses. Instead of having a very disambiguating descriptor, we keep description simple and recover the missing information from HSPA. We use a simple geometrical 6-dimension descriptor called Hexagon (illustration as supplementary material). From a point cloud with normals and BOARD [38] reference frames, the descriptor is computed at each point (no keypoint extraction required) as the six point-to-point distances from the vertices of a fixed-size hexagon in the tangent plane (oriented using to the x-axis of the reference frame) to the object's surface. Normal orientation (inwards/outwards) gives each of the six distances a sign. Descriptor matching generates correspondences, which along with local referentials create pose hypotheses.

We represent poses in 6D Hough space by the translation components and an angle-axis representation of rotations (angle times unit rotation axis $\theta.\vec{r}/||\vec{r}||$). This representation is

optimally compact (3 parameters for rotations) and the cyclicity of the angular coordinate is much easier to handle than with Euler angles (as used for instance in [32, 33, 37]). In Hough space, the cyclicity comes down to replicating points at $(\theta - 2\pi).\vec{r}/||\vec{r}||$ whenever necessary.

We perform an agglomerative clustering in 6D Hough space to create robust pose hypotheses from non-robust input hypotheses. Agglomeration is performed by doing fixed radius searches in Hough space around each hypothesis, ignoring points with less than a fixed number of neighbors (*noise filtering*) and propagating clusters to the remaining neighbors. For each cluster, a 6-component ponderated average is accumulated together with a cluster weight, as the sum of input pose weights in the cluster (input pose weights are obtained for instance as descriptor similarities and are also used to ponderate the 6-component average). Extra care is taken to ensure that the $\pm\pi$ seam is handled correctly.

Agglomerative clustering returns clusters of arbitrary shape and size while still eliminating outliers (compared to k-means [30] for instance). Also, compared to finding the 6D point with the most neighbors [17], a very small 6D search radius can be used for neighborhood searches, which saves computation time. It should be noted that this clustering step does not take object invariances into account. The goal is to reduce the number of points in Hough space while increasing robustness of the remaining points (as the average of transformations in each cluster). In practice, keeping only the $N = 100$ highest-weight clusters is sufficient.

Pose refinement is then performed on the cluster's average using a simple point-to-point ICP [7], which we found more robust for mostly planar objects than point-to-plane ICP.

The 3D registration pipeline described in this section performs registration of objects without invariances or with a low amount of invariances. It works well for these objects since scene-to-model correspondences produce 6D transforms forming a single small cluster in Hough space for each object instance.

3.2 Invariance pattern analysis for 3D registration

3.2.1 Overview

Objects with invariances or symmetries cause mutiple clusters to exist in Hough space for each object instance in the scene.First, we group them into one meta-cluster per object instance using a Hough-space canonical invariance pattern computed and discretized ("beads" model) offline on the reference object or model. Then, we apply a step called *disambiguation* whose goal is to decide between quasi-symmetries (this should be viewed as a second order registration, refining the first order registration obtained at the previous step). Disambiguation uses a disambiguation map computed offline on the reference object or model. A visual representation of the process can be found in the supplementary material/Figure 4.

3.2.2 The canonical invariance pattern

Obtaining the canonical invariance pattern. We start by performing a registration of the model to a version of itself with noise added. Adding noise is necessary to emulate an object viewed by a sensor and overcome the original discretization of an object represented as a point cloud or mesh. It prevents all descriptors from only matching with themselves. Self-registration is performed offline using the the same registration approach as online: optional keypoints detection, description and matching generate correspondences. Correspondences yield 6D transformations expressed in Hough space: the *canonical invariance pattern*.

Obtaining the discretized bead model. The set of correspondences in Hough space is discretized through fixed radius clustering. We call the centers of the clusters *beads*. The role of this discretization is mainly for efficiency for the following online computations.

3.2.3 Grouping clusters belonging to a single object instance

The output of a registration pipeline such as that of section 3.1 is a set of poses, each pose associated to a set of correspondences (a cluster), a weight and a model identifier, if there are multiple models to localize at the same time. The first step of invariance analysis is to group together poses belonging to the same model instance in the scene. For symmetrical objects, multiple pose hypotheses are valid. For objects with invariances but no symmetries, only one pose is correct, but some parts of the objects may match some parts of the scene. For instance, a plane may align on any plane. The process of pose grouping goes as follows (algorithmic form in supplementary material): for each pose hypothesis P_i with weight w_i aligning model M to the scene, let $\forall j, P'_j = P_i^{-1} P_j$. According to the chain rule, P'_j writes as a transformation from the model to itself, and thus should belong to the canonical invariance pattern. By applying P_i^{-1}, we transform the invariance pattern into what should be the canonical invariance pattern *if P_i is a correct pose hypothesis*. Then, for each P_i and each P_j check if $P'_j = P_i^{-1} P_j$ belongs to the canonical invariance pattern or bead discretization thereof (6D distance lower than a threshold r_s, taking into account the $\pm\pi$ seam). If yes, add its weight (multiplied by the canonical pattern's bead's own weight if using beads) to P_i's weight counter (initialized at zero). Then, take the highest score P_i and attach all compatible P_j to it, increasing the final score of P_i by that of all the P_j attached. Neutralize P_i and its attached P_j and restart the process until all poses have been considered. If poses in Hough space are considered as representatives of an underlying probability distribution, the above process consists in finding the transformation which, when applied to the current invariance pattern, maximizes its correlation with the canonical invariance pattern. Repeating the process allows identifying multiple object instances, one at a time.

This algorithm has at least an $\mathcal{O}(n^2_{hypotheses})$ complexity (potentially higher depending on the indexing structures used for nearest neighbor searches within the beads), which is why we execute it on poses output by our 6D clustering algorithm and not on raw hypotheses obtained from correspondences. Currently, 6D clustering finds a maximum of 100 pose hypotheses before invariance analysis but only 16 are kept after invariance analysis.

3.2.4 The disambiguation map

Motivation. The cluster grouping approach described in the previous paragraphs attaches, let's say, clusters B and C to cluster A. For quasi-symmetrical objects such as the *star* (Figure 1), we can't be sure that A is the best pose hypothesis, because descriptors may not see the small details that would allow disambiguation. So, B or C may be better options than A. However, if we have either A, B or C for the *star*, we know that we could obtain both others by applying a few 30° rotations. More generally, if a pose is almost correct up to a quasi-symmetry, applying one of the beads belonging to the canonical invariance pattern will bring it to a correct pose. Which bead to apply is chosen by computing a feature matching score between model and scene for each bead, possibly ponderated by the *disambiguation map*.

Offline computation. Once beads are computed, we can obtain a disambiguation map for a set of local features. Some such features are contours (or 3D edgelets), points with or

without normals, descriptors, etc. and are typically not the descriptors used to compute the Hough pattern. Let $F = \{f_i\}$ be the set of features and $P = \{P_i\}$ be the set of poses, with each pose being the center of one bead and each feature f_i associated with a 3D position $r(f_i)$ on the model. Also, let ω_i be the weight (sum of correspondence weights in the bead) of P_i. Let M be a model. $P_i(M)$ is the model transformed by P_i. For each $f_i \in F$ on M, let $f_i'^j$ be the feature on $P_j(M)$ that is closest to f_i in terms of position, that is $f_i'^j = f_{argmin_k ||P_j(r(f_k)) - r_i||}$. Let w_i^j be the matching score of f_i and $f_i'^j$. We define the feature disambiguation weight of f_i as $W_i = \frac{\sum_{bead\ j} w_i^j \omega_j}{\sum_{bead\ j} \omega_j}$. In other words, the disambiguation weight of a feature is the sum, for all beads, of how different the feature is from the closest feature in the model transformed by the bead. The map of disambiguation weights of all features constitutes what we call the disambiguation map. An example of disambiguation maps using points as features and the distance to the closest point, ponderated by the absolute cosine of the dot product of normals, as score, is plotted in the supplementary material as Figure 2.

Online usage Let P be the current pose hypothesis and $\{P_i\}$ be the set of beads from the canonical invariance pattern. If the Identity transformation is not in the canonical pattern, we add it nevertheless to the set of P_i. Let $F = \{f_i\}$ be the set of features of the scene, $r(f_i)$ being the position of f_i. Similarly, $F' = \{f'\}$ is the model's feature set. We find the closest model feature to a scene feature for a transformation $P.P_j$ as $f_i'^j = f_{argmin_k ||P.P_j(r(f_k)) - r_i||}$. Let w_i^j be the matching score of f_i and $f_i'^j$. We can define a pose matching score for $P.P_j$ as $S_j = \sum_{i, ||r(f_i) - r(f_i'^j)|| < \tau} w_i^j . W_i$ where W_i is a feature f_i's weight in the disambiguation map and τ is a fixed distance (say, 2mm) corresponding to the maximum allowed discrepancy between registered model and scene. The pose $P.P_j$ leading to the highest S_j is chosen instead of P.

4 Experiments

Our experiments are organized as follows: first we evaluate our method on an object pose estimation task, where the goal is to localize up to N instances of a known object (i.e. we have a 3D model of it) in a scene. For this, we follow the rules of the BOP challenge [23] and focus on the ITODD dataset [11] which is representative of industrial objects (shiny, texture-less, etc.). We then evaluate our approach on a scene registration task in a 3D reconstruction context, which consists in recovering the displacement of a camera between two views of the same scene. There, we use the 3DMatch dataset [46] and its evaluation metrics. Experimental details as well as hyperparameters can be found in the supplementary material.

4.1 6DoF object pose estimation

Evaluation metrics. In the BOP Challenge, average recall (AR) is defined as $\text{AR} = (\text{AR}_{VSD} + \text{AR}_{MSSD} + \text{AR}_{MSPD})/3$, the mean between Visible Surface Discrepancy, Maximum Symmetry-Aware Surface Distance and Maximum Symmetry-Aware Projection Distance [23], and is evaluated through the BOP evaluation server. Note that quasi-symmetrical objects such as "star" (see Figure 1) are considered completely symmetrical in the challenge, so the metrics of the BOP challenge can't completely highlight the benefits of symmetry disambiguation and the disambiguation map as explained in this paper.

Method	Modality	AR	Timing
Drost-CVPR10-Edges [12]	RGB-D	0.570	6.833s
SurfEmb [18]	RGB-D	0.538	4.942s
Koenig-Hybrid-DL-PP combination of [14][19][12]	RGB-D	0.483	0.318s
Drost-CVPR10-3D-Edges [12]	Depth only	0.462	5.838s
Vidal-Sensors18 [44]	Depth only	0.435	3.419s
Drost-CVPR10-3D-Only [12]	Depth only	0.316	2.0s
HSPA (whitout invariance analysis)	Depth only	0.412	0.535s
HSPA (whith invariance but no disambiguation)	Depth only	0.458	**0.471s**
HSPA (with invariance and disambiguation)	Depth only	**0.511**	0.567s

Table 1: HSPA (with ablation study) and current top competitors on BOP/ITODD. We focus on the approaches using only the depth modality (bottom rows) but also give the overall best performers using RGB-D (top rows).

On ITODD (Table 1), the performance of our approach with Hough space clustering but no invariance analysis is average: third in terms of AR for the depth only modality. The computation time is more than six times lower than the two better approaches, showing that hexagon is a decent yet not great descriptor. It is fast to compute and match though, which justifies its use to quickly populate Hough space with many pose hypotheses.

With invariance analysis but no disambiguation, AR improves to almost that of the current best depth-only method (0.458 vs 0.462). Computation time decreases by 12% due to each cluster fused into another cluster being one less hypothesis to refine and score. With both invariance analysis and disambiguation, computation time compared to the no invariance analysis case increases by 6%. The average recall gains 0.099 points, placing the approach first among depth-only methods. When comparing to state of the art approaches that also use RGB, our approach has noticeably better average recall than Koenig-Hybrid-DL-PointPairs while being a little slower (their time gain was obtained by segmenting data first with a convolutional neural network before using Drost-CVPR10-Edges [12], so there is a speed/accuracy trade-off). Compared with the current best approach in terms of AR [12], our approach is $12x$ faster (although not evaluated on the same computer) while loosing 0.059 AR. Table 1 displays the top contenders of the challenge for the depth and RGB-D modality for this dataset. PPF-CVPR-10 (Implementation of HALCON 19.05) is still the best performing method on this dataset, outperforming more recent (including 2021) approaches.

4.2 3D Scene registration

The main difference between 3D scene registration and object localization is that in the latter, the reference model is loaded and processed only once "offline", while scenes are processed online. In scene registration, the "object" is a scene. Since the "beads" algorithms was not designed to be fast, we replace it by the 6D clustering used in the online phase.

Evaluation metrics. Following DGR [10] and PointDSC [6], we use three evaluation metrics, namely (1) Registration Recall (RR), the percentage of scenes that were successfully aligned (rotation error and translation error below some thresholds of the ground truth), (2) standard rotation error (RE), and (3) standard translation error (TE).

Method	RR (% ↑)	RE (° ↓)	TE (*cm* ↓)	Time (s)
RANSAC-100k	73.57	3.55	10.04	5.24/-
3DRegNet [34]	26.31	3.75	9.60	0.05/-
DGR w/o s.g. [10]	27.04	2.61	7.76	0.56/-
DGR [10]	69.13	3.78	10.80	2.49/-
PointDSC [6]	78.50	2.07	6.57	0.09/-
HSPA (no invariance)	75.64	1.79	7.91	0.008/0.089
HSPA (with invariance)	81.99	1.82	7.19	0.009/0.092

Table 2: Comparison with state of the art methods on the 3DMatch dataset, based partially on [6]. Only methods based on FPFH [41] have been used for fair comparison with our naive geometric descriptor hexagon and to avoid overfitting a learnt descriptor to a set of scenes. Time is only given for grouping and invariances (when present), when using keypoints only/all points. RR, RE and TE are computed in our case with all points.

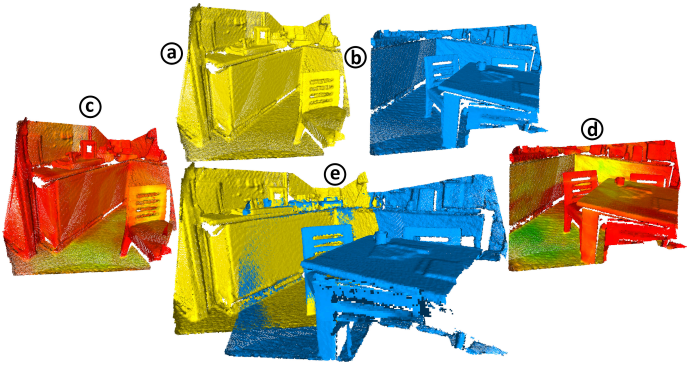

Figure 2: Registration on 3DMatch/redkitchen. (a), (b): input point clouds; (c), (d): computed disambiguation maps, from green (low disambiguation potential) to red (high disambiguation potential); (e) registration. (best viewed in color)

Results. Figure 2 shows a registration between two scenes of "redkitchen" [46]. The floor is never considered highly disambiguating but most high curvature objects are. For (c), we think the chairback was not considered highly disambiguating as in (d) because of a discrete translational invariance which is incomplete due to occlusions in (d). Vertical planes were not considered disambiguating in (d) because there are many other flat surfaces.

Table 2 compares results obtained with hexagon to results obtained with FPFH, another "classical" descriptor. Without invariance analysis, our approach performs similarly to the current best approach PointDSC [6], with similar RR, RE and TE. Invariance analysis allows our approach to take the lead with 82% RR while preserving RE and TE. Computation times are to be taken with caution, as only correspondence processing/pruning time (including invariance analysis when present) is listed to be consistent with [6]. When operating on all points, our computation times are comparable to [6] which is operating on keypoints only. When operating on keypoints only, our computation times decrease by about a factor ten with a 4% RR decrease. Also, hexagon computations and matching are feasible on all points because of the low dimensionality of the descriptor, and the computation times would prevent the computation and matching of FPFH on all points in practical situations.

5 Conclusion and perspectives

In this paper, we proposed a new framework to tackle a common issue with 3D registration, where some zones in an object have the same descriptor (an invariance). When matching such descriptors, Hough space patterns appear. These can be precomputed automatically for any object and matched when trying to find the object in a scene. The Hough pattern thus becomes a global object descriptor, as opposed to local descriptors. We also exposed algorithms to automatically find which parts of an object break the object's common symmetry if any. This disambiguating information can be used in a second pass to refine registration results, which is necessary for objects with fool-proofing or asymmetric holes for instance. We demonstrated state of the art performance in object localization and scene registration benchmarks. Conceptually, our method should be agnostic to the descriptor choices. However, the practical balance between local description power and hough space invariance pattern needs an in-depth investigation. We are currently investigating extensions of invariance analysis to other parametric problems.

References

[1] Sergey V. Alexandrov, Timothy Patten, and Markus Vincze. Leveraging symmetries to improve object detection and pose estimation from range data. In Dimitrios Tzovaras, Dimitrios Giakoumis, Markus Vincze, and Antonis Argyros, editors, *Computer Vision Systems*, pages 397–407, Cham, 2019. Springer International Publishing. ISBN 978-3-030-34995-0.

[2] Sk Aziz Ali, Kerem Kahraman, Gerd Reis, and Didier Stricker. Rpsrnet: End-to-end trainable rigid point set registration network using barnes-hut 2d-tree representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13100–13110, 2021.

[3] Sheng Ao, Qingyong Hu, Bo Yang, Andrew Markham, and Yulan Guo. Spinnet: Learning a general surface descriptor for 3d point cloud registration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11753–11762, 2021.

[4] Yasuhiro Aoki, Hunter Goforth, Rangaprasad Arun Srivatsan, and Simon Lucey. Pointnetlk: Robust & efficient point cloud registration using pointnet. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7163–7172, 2019.

[5] Xuyang Bai, Zixin Luo, Lei Zhou, Hongbo Fu, Long Quan, and Chiew-Lan Tai. D3feat: Joint learning of dense detection and description of 3d local features. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6359–6367, 2020.

[6] Xuyang Bai, Zixin Luo, Lei Zhou, Hongkai Chen, Lei Li, Zeyu Hu, Hongbo Fu, and Chiew-Lan Tai. Pointdsc: Robust point cloud registration using deep spatial consistency. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15859–15869, 2021.

[7] P. J. Besl and N. D. McKay. A method for registration of 3-d shapes. 14(2):239–256, 2 1992. ISSN 0162-8828. doi: 10.1109/34.121791.

[8] Romain Brégier, Frédéric Devernay, Laetitia Leyrit, and James L. Crowley. Symmetry aware evaluation of 3d object detection and pose estimation in scenes of many parts in bulk. In *Proceedings of the 2017 IEEE International Conference on Computer Vision Workshop (ICCVW)*, pages 2209–2218, 2017. doi: 10.1109/ICCVW.2017.258.

[9] Romain Brégier, Frédéric Devernay, Laetitia Leyrit, and James L. Crowley. Defining the pose of any 3d rigid object and an associated distance. *International Journal of Computer Vision*, 126(6):571–596, June 2018. ISSN 1573-1405. doi: 10.1007/s11263-017-1052-4.

[10] Christopher Choy, Wei Dong, and Vladlen Koltun. Deep global registration. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2514–2523, 2020.

[11] B. Drost, M. Ulrich, P. Bergmann, P. Härtinger, and C. Steger. Introducing mvtec itodd - a dataset for 3d object recognition in industry. In *2017 IEEE International Conference on Computer Vision Workshops (ICCVW)*, pages 2200–2208, Oct 2017. doi: 10.1109/ICCVW.2017.257.

[12] Bertram Drost, Markus Ulrich, Nassir Navab, and Slobodan Ilic. Model globally, match locally: Efficient and robust 3d object recognition. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2010.

[13] Mohamed El Banani, Luya Gao, and Justin Johnson. Unsupervisedr&r: Unsupervised point cloud registration via differentiable rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7129–7139, 2021.

[14] Cheng-Yang Fu, Mykhailo Shvets, and Alexander C Berg. Retinamask: Learning to predict masks improves state-of-the-art single-shot detection for free. *arXiv preprint arXiv:1901.03353*, 2019.

[15] Y. Guo, M. Bennamoun, F. A. Sohel, J. Wan, and M. Lu. 3d free form object recognition using rotational projection statistics. In *2013 IEEE Workshop on Applications of Computer Vision (WACV)*, pages 1–8, Jan 2013.

[16] Y. Guo, M. Bennamoun, F. Sohel, M. Lu, and J. Wan. 3d object recognition in cluttered scenes with local surface features: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(11):2270–2287, Nov 2014. ISSN 0162-8828. doi: 10.1109/TPAMI.2014.2316828.

[17] Yulan Guo, Ferdous Sohel, Mohammed Bennamoun, Min Lu, and Jianwei Wan. Rotational projection statistics for 3d local surface description and object recognition. *International Journal of Computer Vision*, 105(1):63–86, 10 2013. ISSN 1573-1405. doi: 10.1007/s11263-013-0627-y.

[18] Rasmus Laurvig Haugaard and Anders Glent Buch. Surfemb: Dense and continuous correspondence distributions for object pose estimation with learnt surface embeddings. *arXiv preprint arXiv:2111.13489*, 2021.

[19] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.

[20] S. Hinterstoisser, S. Holzer, C. Cagniart, S. Ilic, K. Konolige, N. Navab, and V. Lepetit. Multimodal templates for real-time detection of texture-less objects in heavily cluttered scenes. In *2011 International Conference on Computer Vision*, pages 858–865, Nov 2011. doi: 10.1109/ICCV.2011.6126326.

[21] Stefan Hinterstoisser, Vincent Lepetit, Naresh Rajkumar, and Kurt Konolige. Going further with point pair features. In *Proceedings of the 14th European Conference on Computer Vision (ECCV 2016)*, pages 834–848, 2016. doi: 10.1007/978-3-319-46487-9_51.

[22] Tomás Hodan, Dániel Baráth, and Jiri Matas. EPOS: estimating 6d pose of objects with symmetries. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 11700–11709. IEEE, 2020. doi: 10.1109/CVPR42600.2020.01172.

[23] Tomáš Hodaň, Martin Sundermeyer, Bertram Drost, Yann Labbé, Eric Brachmann, Frank Michel, Carsten Rother, and Jiří Matas. Bop challenge 2020 on 6d object localization. In *European Conference on Computer Vision*, pages 577–594. Springer, 2020.

[24] T. Hodaň, X. Zabulis, M. Lourakis, S. Obdržálek, and J. Matas. Detection and fine 3d pose estimation of texture-less objects in rgb-d images. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4421–4428, Sep. 2015. doi: 10.1109/IROS.2015.7354005.

[25] Shengyu Huang, Zan Gojcic, Mikhail Usvyatsov, Andreas Wieser, and Konrad Schindler. Predator: Registration of 3d point clouds with low overlap. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4267–4276, 2021.

[26] Xiaoshui Huang, Guofeng Mei, Jian Zhang, and Rana Abbas. A comprehensive survey on point cloud registration. *arXiv preprint arXiv:2103.02690*, 2021.

[27] Kourosh Khoshelham. Extending generalized hough transform to detect 3d objects in laser range data. In *Proceedings of the ISPRS Workshop on Laser Scanning 2007 and SilviLaser 2007*, 2007.

[28] C. Li, J. Bai, and G.D. Hager. A unified framework for multi-view multi-class object pose estimation. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 11220:263–281, 2018. doi: 10.1007/978-3-030-01270-0_16.

[29] Fabian Manhardt, Diego Martin Arroyo, Christian Rupprecht, Benjamin Busam, Tolga Birdal, Nassir Navab, and Federico Tombari. Explaining the ambiguity of object detection and 6d pose from visual data. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 6841–6850, 2019.

[30] A. Mian, M. Bennamoun, and R. Owens. On the repeatability and quality of keypoints for local feature-based 3d object retrieval from cluttered scenes. *International Journal of Computer Vision*, 89(2):348–361, Sep 2010. ISSN 1573-1405. doi: 10.1007/s11263-009-0296-z.

[31] A. S. Mian, M. Bennamoun, and R. Owens. Three-dimensional model-based object recognition and segmentation in cluttered scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(10):1584–1601, Oct 2006. ISSN 0162-8828. doi: 10.1109/TPAMI.2006.213.

[32] Niloy J. Mitra, Leonidas J. Guibas, and Mark Pauly. Partial and approximate symmetry detection for 3d geometry. *ACM Trans. Graph.*, 25(3):560–568, July 2006. ISSN 0730-0301. doi: 10.1145/1141911.1141924.

[33] Niloy J. Mitra, Mark Pauly, Michael Wand, and Duygu Ceylan. Symmetry in 3d geometry: Extraction and applications. *Computer Graphics Forum*, 32(6):1–23, 2013. doi: 10.1111/cgf.12010.

[34] G Dias Pais, Srikumar Ramalingam, Venu Madhav Govindu, Jacinto C Nascimento, Rama Chellappa, and Pedro Miraldo. 3dregnet: A deep neural network for 3d point registration. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7193–7203, 2020.

[35] Chavdar Papazov, Sami Haddadin, Sven Parusel, Kai Krieger, and Darius Burschka. Rigid 3d geometry matching for grasping of known objects in cluttered scenes. *International Journal of Robotics Research*, 31(4):538–553, April 2012. ISSN 0278-3649. doi: 10.1177/0278364911436019.

[36] Kiru Park, Timothy Patten, and Markus Vincze. Pix2pose: Pixel-wise coordinate regression of objects for 6d pose estimation. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, Oct 2019. doi: 10.1109/iccv.2019.00776.

[37] Mark Pauly, Niloy J. Mitra, Johannes Wallner, Helmut Pottmann, and Leonidas J. Guibas. Discovering structural regularity in 3d geometry. In *ACM SIGGRAPH 2008 Papers*, SIGGRAPH '08, New York, NY, USA, 2008. Association for Computing Machinery. ISBN 9781450301121. doi: 10.1145/1399504.1360642. URL https://doi.org/10.1145/1399504.1360642.

[38] A. Petrelli and L. Di Stefano. On the repeatability of the local reference frame for partial shape matching. In *2011 International Conference on Computer Vision*, pages 2244–2251, Nov 2011. doi: 10.1109/ICCV.2011.6126503.

[39] Giorgia Pitteri, Michaël Ramamonjisoa, Slobodan Ilic, and Vincent Lepetit. On object symmetries and 6d pose estimation from images. In *2019 International Conference on 3D Vision (3DV)*, pages 614–622, 09 2019. doi: 10.1109/3DV.2019.00073.

[40] Mahdi Rad and Vincent Lepetit. Bb8: A scalable, accurate, robust to partial occlusion method for predicting the 3d poses of challenging objects without using depth. In *Proceedings of the International Conference on Computer Vision (ICCV 2017)*, 2017. URL https://arxiv.org/abs/1703.10896.

[41] R. B. Rusu, N. Blodow, and M. Beetz. Fast point feature histograms (fpfh) for 3d registration. In *2009 IEEE International Conference on Robotics and Automation*, pages 3212–3217, May 2009. doi: 10.1109/ROBOT.2009.5152473.

[42] F. Tombari and L. Di Stefano. Object recognition in 3d scenes with occlusions and clutter by hough voting. In *Fourth Pacific-Rim Symposium on Image and Video Technology*, pages 349–355, Nov 2010. doi: 10.1109/PSIVT.2010.65.

[43] Federico Tombari, Samuele Salti, and Luigi Di Stefano. *Unique Signatures of Histograms for Local Surface Description*, pages 356–369. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010. ISBN 978-3-642-15558-1. doi: 10.1007/978-3-642-15558-1_26.

[44] Joel Vidal, Chyi-Yeu Lin, Xavier Lladó, and Robert Martí. A method for 6d pose estimation of free-form rigid objects using point pair features on range data. *Sensors*, 18(8):2678, 2018.

[45] Jiaqi Yang, Ke Xian, Peng Wang, and Yanning Zhang. A performance evaluation of correspondence grouping methods for 3d rigid data matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(6):1859–1874, June 2021. ISSN 1939-3539. doi: 10.1109/TPAMI.2019.2960234.

[46] Andy Zeng, Shuran Song, Matthias Nießner, Matthew Fisher, Jianxiong Xiao, and Thomas Funkhouser. 3dmatch: Learning local geometric descriptors from rgb-d reconstructions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1802–1811, 2017.

[47] Y. Zhong. Intrinsic shape signatures: A shape descriptor for 3d object recognition. In *2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops*, pages 689–696, Sept 2009. doi: 10.1109/ICCVW.2009.5457637.