Supplementary Material: 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

1 Algorithms

Algorithm 1 does invariance-based correspondence grouping as mentionned in the paper, section 3.2.3. It essentially applies each pose hypothesis in turn to bring the scene-to-model Hough pattern to a model-to-model pattern and compares it to the canonical (model-to-model) pattern.

2 Experimental protocol and parameters

Figure 1 (right) describes the main steps of our registration pipeline. Each of the steps of the Figure is in practice made of many smaller steps. We would like to emphasize the fact that although this pipeline is classical and well mastered in literature, its implementation makes a huge difference in terms of computation time as well as precision and recall of the results. Each step needs to be carefully tuned for optimal performance. In the following paragraphs, we try to reflect most implementation details that may be relevant for reproducing our results.

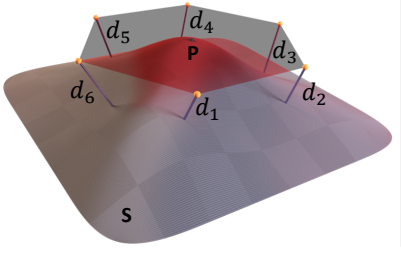
First of all, 3D nearest neighbor searches (searches in a given radius or searches for the n-nearest neighbors) are backed by a custom GPU kd-tree implementation. Scene-to-model first nearest neighbor searches (as used in ICP) are backed by a GPU voxel discretization of the model for speed.

2.1 Model and scene preparation

First, when performing CAD model registration, the CAD model is prepared by computing normals and reference frames.

Algorithm 1 Correspondence grouping algorithm

pose hypotheses $P = \{P_i \text{ with weight } w_i\}$ aligning model M to the scene
$C_i, v_i \leftarrow$ canonical pattern with weight $v_i = 1$ for each pose C_i, or weighted bead discretization thereof
$M \leftarrow \emptyset$
while $P \neq \emptyset$ **do**
    $A \leftarrow [\emptyset \dots \emptyset]$
    $W \leftarrow [0 \dots 0]$
    for $(P_i, w_i) \in P$ **do**
        $W[i] \leftarrow 0$
        for $(P_j, w_j) \in P$ **do**
            $P_{ij} \leftarrow P_i^{-1} P_j$ ▷ P_{ij} is a model-to-model transformation (chain rule) and should belong to the canonical invariance pattern. Applying P_i^{-1} transforms the current invariance pattern into the canonical one IF P_i is correct
            if $D(P_{ij}, C_k) < r_s$ with $C_k = argmin_l(D(P_{ij}, C_l))$ **then** ▷ D takes into account the $\pm\pi$ seam
                $W[i] \leftarrow W[i] + w_i.w_j.v_k, A[i] \leftarrow A[i] \cup j$
            end if
        end for
        $(P_m, W_m) \leftarrow argmax_{W[i]}\{(P_i, W[i])\}$ ▷ select the P_i with highest combined weight
        $M \leftarrow P' \cup (P_m, W_m)$, $P \leftarrow P \setminus \{(P_{A[i]}, w_{A[i]})\}$ ▷ remove selected P and its attached transformations from pool
    end for
end while
    return M ▷ fused transformations with weight



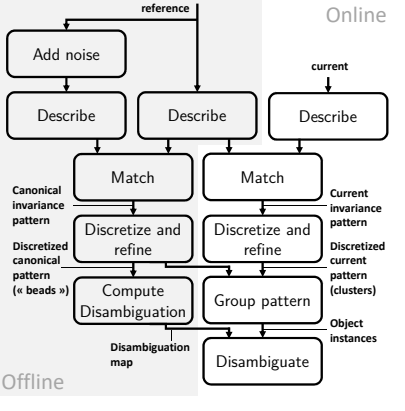


Figure 1: Left: Hexagon, a simple geometric descriptor, obtained by sampling six signed distances $d_1 - d_6$ from the tangent plane to the objects' surface S. Right: Our invariance analysis process for 3D registration (details in the supplementary material). One match with reference frames, two matches with normals or three with just points give a 6DOF pose hypothesis. For point descriptors, we compute one descriptor per 3D point or keypoint and generate one match for each.

Normals are computed at point X as the eigen vector associated to the lowest magnitude eigenvalue of $(X_i - X).\bar{(}X_i - X)^T - X_i \bar{-} X.X_i \bar{-} X^T$ where the X_i are all points in a fixed size neighborhood (*normal computation radius* r_n) of the computation point. Since the both n and $-n$ are valid eigenvectors, we force normal orientation depending on curvature, so that normals of a convex object are oriented outwards. This is done by finding the sign of $n \cdot (X_i \bar{-} X - Y)$ where Y is an average of local points with non-unitary ponderation: $Y = \sum_i \frac{1}{(1+||X-X_i||/r_n}(X_i - X)/\sum_i \frac{1}{(1+||X-X_i||/r_n}$. For planar surfaces, normals can go either way.

Reference frames are computed using neighboring points in a radius r_{RF} with a tweaked version of BOARD [38] where the border detection algorithm is kept but when a border is detected, the normal is always oriented away from the barycenter of local points (and thus towards the border). This is necessary to ensure that completely flat objects can be described. Indeed, the original implementation of BOARD [38] fails with borders around planar surfaces.

Description with hexagon (Figure 1 (left)) is simply performed by sampling six point-to-point distances as described in the main paper. The radius of the hexagon is r_{hex}.

When preparing scenes (as opposed to CAD models), a Moving Least Squares (MLS) smoothing is first applied with radius r_{MLS} to filter out noise (such as the "eggbox" effect observed on many 3D sensors) and remove isolated points. Smoothing helps the hexagon descriptor being repeatable since it is based on sampling few (6) distances and is thus not intrinsically robust to noise, compared to, for instance, FPFH [41] or SHOT [42].

For ITODD [11] specifically, automatic plane suppression (using 1000 RANSAC iterations) was used since ITODD is a dataset of objects lying on a plane.

2.2 Matching

Correspondences are found from scene to model by finding, for each descriptor of the scene, its closest match (n-closest matches could also be used) in the model. Descriptor matches whose discrepancy (for descriptors normalized to one) exceeds the *descriptor mismatch threshold* are ignored.

One correspondence from local reference frame to local reference frame gives a pose hypothesis, which we used in the paper. Two correspondences from point with normal to point with normal also give a pose hypothesis, which can be used when no reference frames are available. Finally, three correspondences from point to point also yield a pose hypothesis.

2.3 Hough-space processing

Pose hypotheses are rigid 6DOF transformations expressed as a 3-vector translation and angle axis rotation. The translation part is scaled by $1/t_s$ where t_s is the model's bounding box maximum size. The rotation part is scaled by $1/\pi$. Agglomerative clustering is performed with a nearest neighbor search radius r_6 and discards points with less than N_6 neighbors.

Canonical invariance patterns are computed using r_6 as bead radius and target a h_c coverage of all poses with a maximum of N_b beads. Random uniform 3D noise of magnitude δ_n is added for self-matching.

Cluster grouping using invariance analysis is performed with 6D radius ri_6.

Figure 2 shows disambiguation maps for some objects.

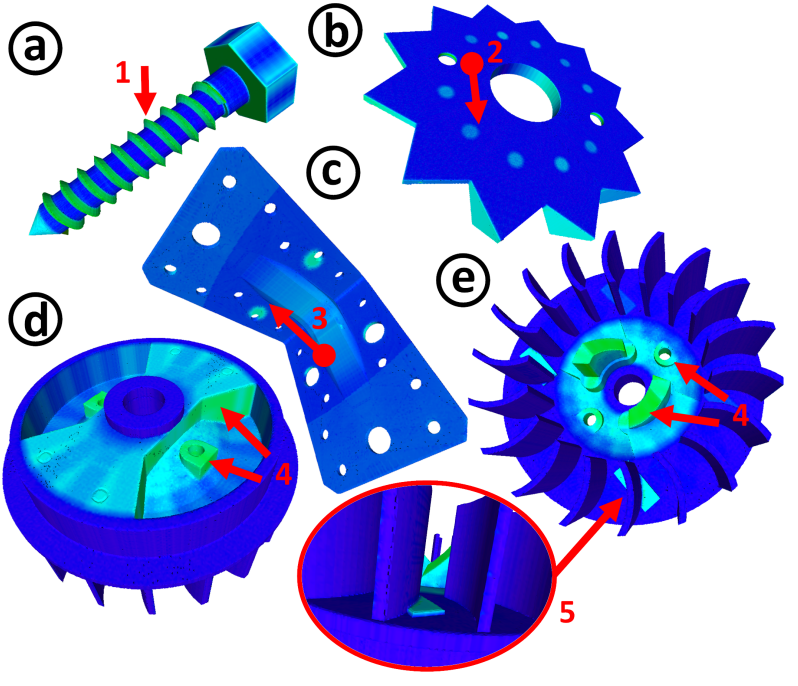


Figure 2: Disambiguation maps colored from blue to green according to the disambiguation potential of individual points (best viewed in color). a) the screw's threading is disambiguating while the cylinder forming the body of the screw is not (1). b) the little holes form a repeating pattern (2) where they would be located if 30° rotations were to occur. c) the same phenomenon occurs (3) because the 180° rotation invariance is broken by two holes of different sizes on each side. d) and e) are two points of view of the same object. Here, disambiguating details get highlighted (4). In case (5), the highlighted zone is very small compared to other features of the object. We did not notice it on the CAD of the object until symmetry disambiguation highlighted it.

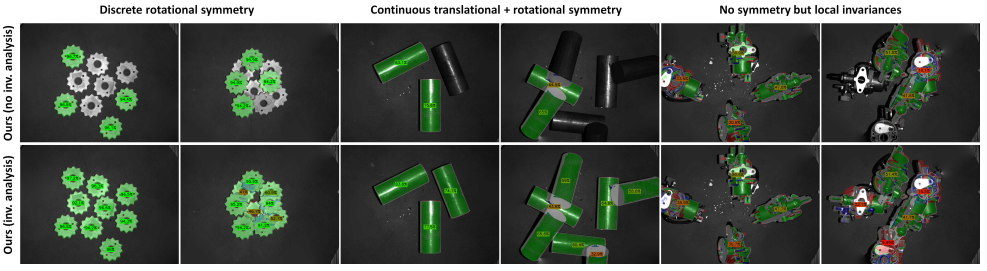


Figure 3: Registration examples using depth only. Top: baseline (our method without invariance analysis). Bottom: proposed method with invariance analysis, with confidence scores (highlighted in red to green depending on confidence). Three objects from the ITODD [11] dataset are illustrated: "star" with a discrete rotational symmetry, "cylinder" with a continous rotational symmetry and "pump" without global symmetries but with local invariances for the hexagon descriptor (best viewed in color)

2.4 Refinement and scoring

Pose refinenement is performed using N_{ICP} steps of point-to-point ICP with voxel indexing. Only points below r_{ICP} between scene and model are considered in the ICP.

For scoring (for disambiguation and to produce the final confidence score and sort the poses), an ICP score is computed (average of nearest neighbor distances ponderated by normal discrepancy). When the scene was acquired with a single sensor from a single point of view (as with the ITODD dataset), we restrain the ICP score to visible points only, ignoring occluded points (visible points are computed by rendering the CAD model in a given pose using the Vulkan API). Also, parts of the model located in front of scene points from the sensor point of view are physically impossible (the sensor can't see through objects), so these points contribute negatively instead of positively to the overall ICP score. Figure 3 shows registrations on ITODD [11] with confidence scores obtained by ICP.

2.5 Metaparameter tables

For both BOP/ITODD [23,11] and 3DMatch [46], parameters were chosen heuristically in accordance with described object size and sensor noise level. Automatically computed values were then manually refined to balance precision/recall on the one side and computation time on the other side. Parameters are displayed in table 1 and 2

3 Hough space grouping illustrations

Figure 4 is a visual illustration of the invariance analysis process which illustrates the need for the two invariance anlysis steps: cluster grouping and disambiguation. For "star" (a), the canonical invariance pattern (b) is a single point in translation space and a set of 12 points in rotation space, corresponding to discrete 30° rotations along the start's axis. Among these, only 0° and 180° are perfect symmetries, the other are only good up to the two small holes.

Parameter name	Unit	Value
normal radius r_n, BOARD radius r_{RF} and MLS radius r_{MLS}	mm	2
hexagon radius r_{hex}	mm	10
6D clustering radius r_6	-	0.025
6D outlier rejection min neighbors N_6	-	4
target bead coverage of Hough poses h_c	%	95
max number of beads N_b	-	200
added random noise δ_n	mm	1
cluster grouping radius ri_6	-	0.14
number of ICP steps N_{ICP}	-	200
ICP maximum distance r_{ICP}, τ	mm	2

Table 1: Parameters for ITODD processing.

Parameter name	Unit	Value
normal radius r_n	cm	4
BOARD radius r_{RF}	cm	7
MLS radius r_{MLS}	cm	not used
hexagon radius r_{hex}	cm	10
6D clustering radius r_6	-	0.015
6D outlier rejection min neighbors N_6	-	4
target bead coverage of Hough poses h_c	%	95
max number of beads N_b	-	100
added random noise δ_n	cm	1
cluster grouping radius ri_6	-	0.14
number of ICP steps N_{ICP}	-	100
ICP maximum distance r_{ICP}, τ	cm	5

Table 2: Parameters for 3DMatch processing.

When registering model (a) on scene (c), the first step of HSPA, cluster grouping, takes the raw clusters created by agglomerative 6D clustering ((d)1), (d)3)) and creates one meta-cluster per object instance in the scene. Each meta-cluster results in a single point in translation space ((d)4)) but a set of beads along a curve in rotation space ((d)2)). Each meta-cluster is a warped version of the canonical invariance pattern (b).

The role of disambiguation is to find which bead along the curve is the right one (or in the case of a perfect symmetry, find one of the possible beads. For, say, a cylinder or sphere, any bead is correct). On the Figure 4, from an initial hypothesis ((f)1)) for each meta-cluster, disambiguation chose a position along the pattern ((f)2)) which corresponds to the best rotation of the "star", with aligned small holes. We did not show the translation part of Hough space for disambiguation since nothing interesting happens there (the canonical invariance pattern of "star" does not exhibit translational invariances).

4 HSPA usage and limitations

As visible on Figure 3, the invariance analysis method is well suited for objects presenting invariances. For non-ambiguous objects (the easy case), the method has less interest as it adds computation time (typically about 6%) without providing much benefits. However, the invariance pipeline does never degrade the quality of registration, and as such can be used by default when it is unknown whether or not objects have invariances. An object such as the “pump” of Figure 3 has no global symmetries and yet it benefits from invariance analysis because of its constant curvature (planes/cylinders) parts.

Also, since symmetry disambiguation looks for small details (some very close in size to the resolution of common 3D sensors), using an average nearest neighbor distance or other 3D metrics requires perfectly regular 3D sampling and is not always robust, for instance on surfaces whose normal is far from colinear to the sensor’s optical axis (for 3D sensors, xy pitch is proportional to the distance to the sensor, so such surfaces show large sampling discrepancies). We are not aware of a dataset evaluating disambiguation of small details.

Finally, invariance analysis works best in scenarios where it is possible to generate many pose hypotheses. When only a handful are generated, it may be necessary to adapt the invariance analysis process and its robustness may be questioned. If only a single hypothesis is generated (as may be the case when using neural networks with a final non-maximal suppression layer), the invariance analysis pipeline as described in this paper can’t be used.

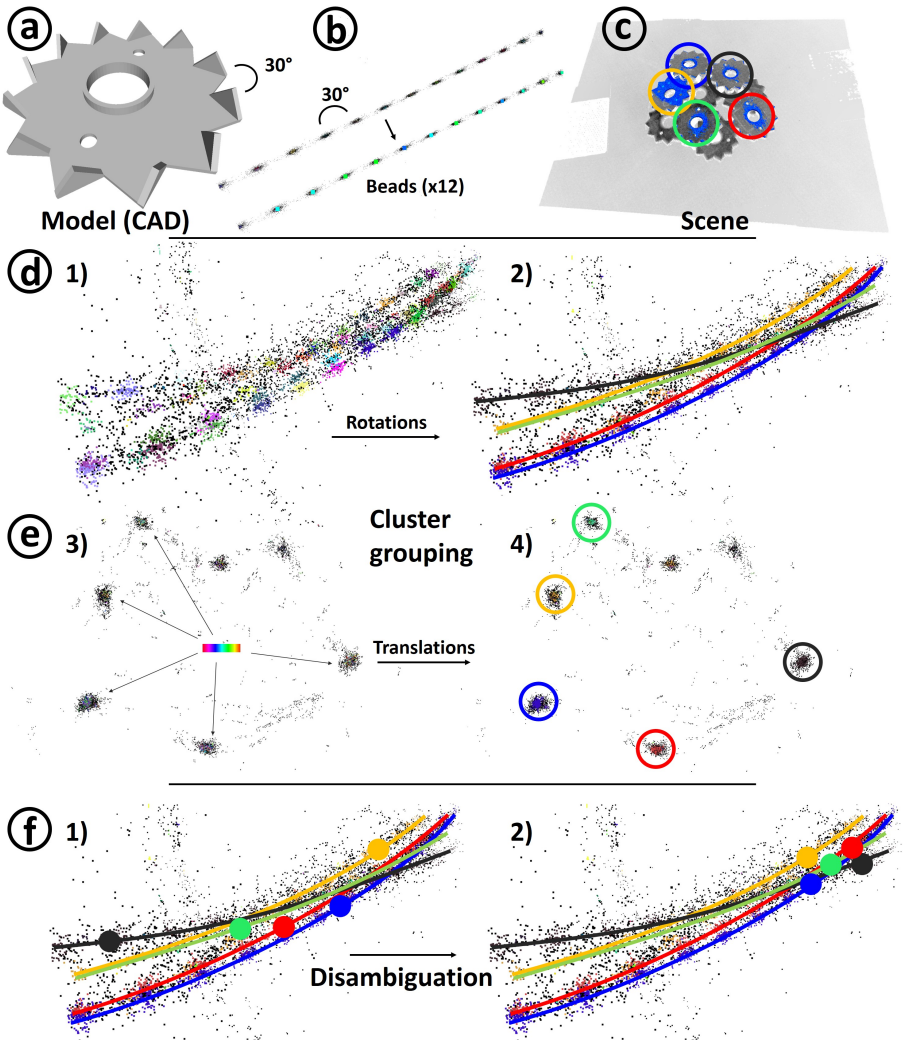


Figure 4: (a) Model. (b) canonical invariance pattern, rotations (translations carry no information). (c) scene. (d,e) invariance analysis groups clusters per object instance. (f) disambiguation finds the correct 30° rotation.