KASALv2: Fully Automatic 3D Rotational Symmetry Classification and Axis Localization

Supplementary Material

This supplementary document provides additional details, analyses, and implementation notes that complement the main paper.

Section 7 presents our geometry-based re-annotation of rotational symmetry in the BOP datasets. It outlines the motivation for re-labeling, the manual geometry-based protocol (Sec. 7.1), the corrected symmetry annotations (Tab. 7), and the accuracy obtained when treating the original BOP labels as ground truth (Tab. 5). Section 8 provides the operational form of the Geometric Degeneration Hierarchy (GDH) used in KASALv2 and specifies the decision rules involved in symmetry-type determination and axis reconstruction. Section 9 offers a detailed analysis of all failure modes, separating type-level and order-level errors and illustrating each category with representative examples.

Section 10 summarizes additional methodological and experimental details, including the localization accuracy e_{ADI}/d computed without excluding misclassified objects (Tab. 13), the rationale behind our choice of loss functions, a direct comparison against the method of Wang et al. (Tab. 12), the implementation procedure for texture-aware symmetry analysis (Sec. 10.2), the training setup for FoundationPose (Sec. 10.3), and notes on hardware, runtime, and code release (Sec. 10.4).

7. Re-Annotation Protocol and Dataset Corrections

7.1. Geometry-Based Re-Annotation of BOP Models

The symmetry annotations provided in BOP were designed to support pose estimation and tracking and therefore emphasize pose-equivalence classes rather than geometric structure. From the standpoint of geometric symmetry analysis, two systematic discrepancies arise: objects with finite repeated features may be annotated with continuous types, and shapes containing asymmetric or non-periodic elements may still be labeled as rotationally symmetric. As a result, the original annotations are not directly suitable as ground truth for geometry-level evaluation in our task.

To obtain labels consistent with geometric criteria, we re-annotate all BOP objects originally marked as rotationally symmetric through direct inspection of their surface geometry. Each mesh is loaded, normalized, and examined from multiple viewing directions to determine whether rotational symmetry is present. When symmetry exists, the rotational order n is identified from countable repeated features; otherwise, the object is labeled as non-symmetric. The resulting geometry-based annotations serve as the ground truth used throughout our evaluation.

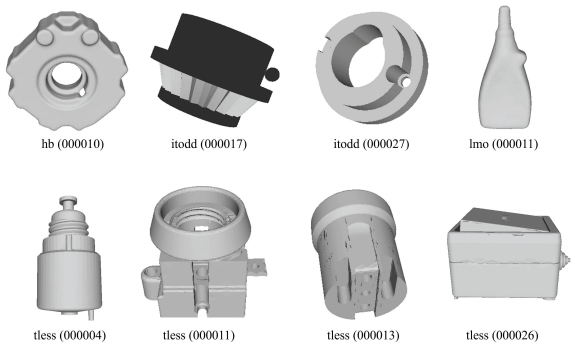

Figure 5. Objects that were originally labeled as rotationally symmetric in BOP but exhibit no valid global rotational symmetry under geometric inspection. These 8 models are excluded from the symmetric set in our re-annotation.

7.2. Summary of Corrections and Non-Symmetric Cases

Based on the original metadata of the seven core BOP datasets, TUDL contains no rotationally symmetric objects, and the remaining six datasets list 58 objects as symmetric. Our geometry-based re-annotation identifies 8 of these as non-symmetric because they lack valid global rotational invariance. Representative cases in Fig. 5 illustrate typical causes, including uneven cross-sectional geometry, asymmetric attachments, and local structures that disrupt periodicity. Although originally labeled as C_∞ or C_n, these objects do not satisfy geometric rotational symmetry and therefore cannot serve as ground truth for type, order, or axis evaluation.

Among the remaining objects, 10 exhibit inconsistencies between the original BOP labels and their true symmetry, as shown in Fig. 6. These shapes possess finite discrete orders, yet the metadata assigns a higher or continuous label such as C_∞. Our corrected annotations provide the geometrically valid types and orders used in all evaluations. Table 4 summarizes the overall effect of our re-annotation: many objects previously labeled as C_∞ are reassigned to finite discrete families, resulting in a final set of 50 geometrically valid rotationally symmetric models. A full comparison between the original and corrected annotations is provided in Table 7, and the accuracy of KASALv2 with respect to the original BOP labels is reported in Table 5.

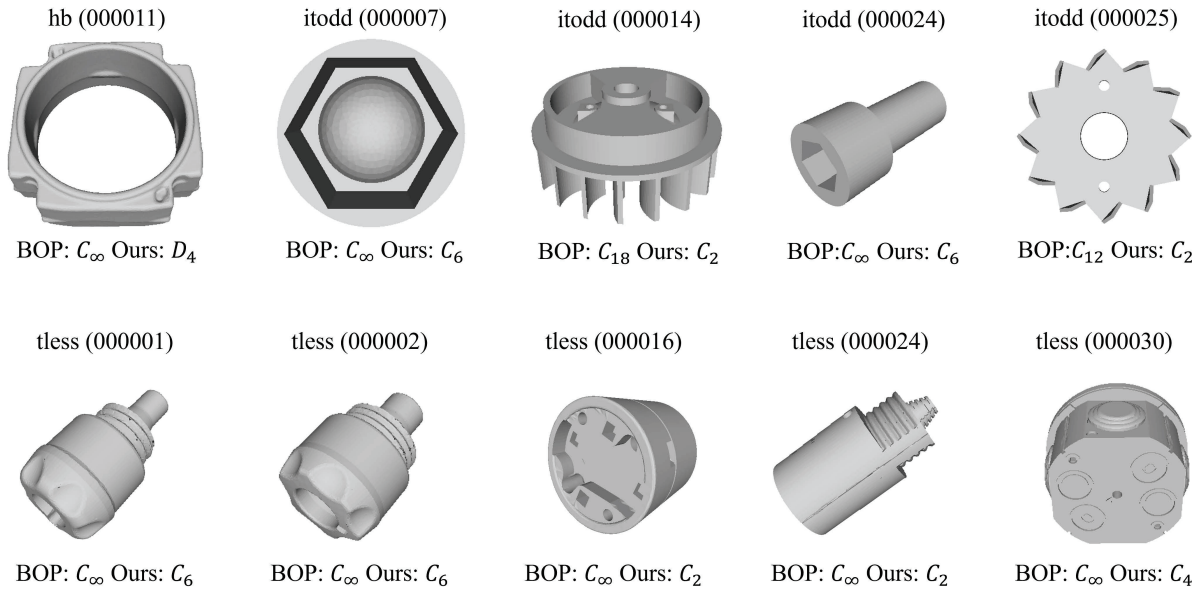

Figure 6. Objects whose original BOP symmetry labels differ from our geometry-based re-annotation. Each model exhibits finite discrete symmetry despite being labeled as continuous or higher-order in the original metadata.

Table 4. Symmetry-type distribution of BOP objects before and after re-annotation.

	C_∞	D_∞	C_n	D_n	Total
Original	19	3	32	4	58
Re-annotated	7	3	35	5	50

Table 5. Accuracy with respect to the original BOP labels.

Type	C_∞	D_∞	C_n	D_n	Total
Count	15	3	28	4	50
Errors	7	0	3	0	10
acc_T	53.33%	100.00%	89.29%	100.00%	80.00%

Table 6. Two-axis GDH criteria for distinguishing rotational symmetry families.

Family	$n_{\max}$	α	n_2
C_∞	∞	—	—
D_∞	∞	90°	2
C_n	n	—	—
D_n	n	90°	2
A_4	3	70.53°	3
S_4	4	90.00°	4
A_5	5	63.43°	5

8. From Sphere to Object: A GDH-Based Symmetry Decision Rule

To complement the brief conceptual introduction in the main paper, we provide here an operational view of GDH that directly connects degeneration levels to observable geometric cues.

8.1. Degeneration Model and Observable Symmetry Signals

In KASALv2, we adopt an operational interpretation of the Geometric Degeneration Hierarchy (GDH): an unknown object is modeled as the result of progressively degrading a perfectly isotropic enclosing sphere into its current form. The sphere serves as a unified root of the hierarchy and can be regarded as an extreme instance of the continuous dihedral family D_∞ in which every direction behaves as a 2-fold axis. Under this perspective, any observable rotational symmetry corresponds to stopping this degeneration at a particular level. Type determination thus amounts to identifying which GDH level is consistent with the geometric evidence measured on the object.

Each symmetry family occupies a unique position in the GDH and is characterized by a fixed pair of observable geometric quantities: the order of the dominant high-order axis, and the order and theoretical relative angle of the second-highest axis when it exists. As the sphere degenerates, the infinite axis set condenses into a dominant axis whose order $n_{\max}$ coarsely distinguishes continuous from finite symmetries and cyclic from dihedral structures. Families that admit a second high-order axis impose strict constraints on its angle α relative to the dominant axis and on its discrete order n_2. These signatures, summarized in Table 6, uniquely identify each family and constitute the geometric criteria that separate the GDH branches.

Table 7. Comparison of original BOP rotational-symmetry labels and our geometry-based re-annotations. Part (a) lists objects from *hb*, *icbin*,*itodd*, *lom*, and *tless*; part (b) lists objects from *tless* and *ycbv*.

(a) part1

Dataset	Object ID	BOP Type	BOP Order	Re-Annotated Type	Re-Annotated Order	Status
hb	000010	C_∞	2	—	—	non-symmetric
	000011	C_∞	2	D_n	4	corrected
	000014	C_∞	2	C_∞	2	consistent
icbin	000002	C_n	2	C_n	2	consistent
itodd	000002	C_n	3	C_n	3	consistent
	000003	D_n	2	D_n	2	consistent
	000004	C_n	2	C_n	2	consistent
	000005	C_n	2	C_n	2	consistent
	000007	C_∞	2	C_n	6	corrected
	000008	C_n	5	C_n	5	consistent
	000009	C_n	2	C_n	2	consistent
	000011	D_n	2	D_n	2	consistent
	000012	D_∞	2	D_∞	2	consistent
	000014	C_n	18	C_n	2	corrected
	000017	C_n	23	—	—	non-symmetric
	000018	C_n	2	C_n	2	consistent
	000019	D_n	2	C_n	2	consistent
	000023	C_n	6	C_n	6	consistent
	000024	C_∞	2	C_n	6	corrected
	000025	C_n	12	C_n	2	corrected
	000027	C_∞	2	—	—	non-symmetric
	000028	D_∞	2	D_∞	2	consistent
lmo	000010	C_n	2	C_n	2	consistent
	000011	C_n	2	—	—	non-symmetric
tless	000001	C_∞	2	C_n	6	corrected
	000002	C_∞	2	C_n	6	corrected
	000003	C_∞	2	C_∞	2	consistent
	000004	C_∞	2	—	—	non-symmetric

(b) part2

Dataset	Object ID	BOP Type	BOP Order	Re-Annotated Type	Re-Annotated Order	Status
tless	000005	C_n	2	C_n	2	consistent
	000006	C_n	2	C_n	2	consistent
	000007	C_n	2	C_n	2	consistent
	000008	C_n	2	C_n	2	consistent
	000009	C_n	2	C_n	2	consistent
	000010	C_n	2	C_n	2	consistent
	000011	C_n	2	—	—	non-symmetric
	000012	C_n	2	C_n	2	consistent
	000013	C_∞	2	—	—	non-symmetric
	000014	C_∞	2	C_∞	2	consistent
	000015	C_∞	2	C_n	2	corrected
	000016	C_∞	2	C_n	2	corrected
	000017	C_∞	2	C_∞	2	consistent
	000019	C_n	2	C_n	2	consistent
	000020	C_n	2	C_n	2	consistent
	000023	C_n	2	C_n	2	consistent
	000024	C_∞	2	C_n	2	corrected
	000025	C_n	2	C_n	2	consistent
	000026	C_n	2	—	—	non-symmetric
	000027	C_n	4	C_n	4	consistent
	000028	C_n	2	C_n	2	consistent
	000029	C_n	2	C_n	2	consistent
	000030	C_∞	2	C_n	4	corrected
ycbv	000001	D_∞	2	D_∞	2	consistent
	000013	C_∞	2	C_∞	2	consistent
	000016	D_n	4	D_n	4	consistent
	000018	C_∞	2	C_∞	2	consistent
	000019	C_n	2	C_n	2	consistent
	000020	C_n	2	C_n	2	consistent
	000021	C_n	2	C_n	2	consistent

Using these criteria, GDH provides an explicit decision process for symmetry classification. Once the dominant axis is localized and its order $n_{\max}$ is estimated, all families inconsistent with $n_{\max}$ are immediately eliminated. As summarized in Table 8, $n_{\max} = \infty$ restricts candidates to C_∞, D_∞, values in $3, 4, 5$ additionally admit the corresponding Platonic family, and all remaining $n_{\max}$ values reduce the search to the cyclic–dihedral pair C_n, D_n. Verifying whether a second high-order axis exists at the prescribed angle α and with the prescribed order n_2 resolves the family unambiguously. The two non-collinear axes then uniquely determine a rigid alignment to the canonical template of that family, enabling full, group-consistent reconstruction of all symmetry axes.

8.2. GDH-Based Search-Space Pruning

For each remaining candidate, the table also specifies the required secondary-axis cue (α, n_2). Types with $\alpha =$ — (all C_n and C_∞) require no further axis search and are verified directly from the dominant-axis test. For the other families, we perform an exact ring search at inclination α around the dominant axis and evaluate the n_2-fold Chamfer-based periodic alignment loss along this ring. The minimum-loss direction is taken as the putative secondary axis, and the candidate is accepted only when its loss satisfies the acceptance condition for an n_2-fold periodic signal.

Table 8. GDH-based pruning rules from the dominant order $n_{\max}$.

$\mathbf{n}_{\max}$	Family	$\boldsymbol{\alpha}$	$\mathbf{n_2}$
∞	C_∞	—	—
	D_∞	90°	2
3	C_3	—	—
	D_3	90°	2
	A_4	70.53°	3
4	C_4	—	—
	D_4	90°	2
	S_4	90°	4
5	C_5	—	—
	D_5	90°	2
	A_5	63.43°	5
other	C_n	—	—
	D_n	90°	2

This procedure converts type classification into a compact two-step decision: (i) pruning by the dominant order via Table 8, followed by (ii) verification of a single secondary axis at the prescribed (α, n_2).

8.3. GDH Canonical Axis Templates (for Full Reconstruction)

Each symmetry family admits a fixed set of canonical axis directions on the unit sphere. Once the dominant high-order axis and any one non-collinear high-order axis have been recovered on the object, a unique rigid transform (up to axis sign) aligns these observed axes with the corresponding canonical pair in the template. This alignment determines the entire axis system without further searching, because all remaining axes are fully specified by the family-specific canonical configuration.

Table 9 lists the complete axis templates used in KASALv2. Each entry provides the full set of unit axis directions for that family, expressed in a principal frame where the dominant axis is $\hat{z}$. Undirected axes are written in $\pm v$ form, and "order:count" indicates the rotational order of each axis and its multiplicity. The constant $\phi = (1+\sqrt{5})/2$ appears in the icosahedral construction.

Table 9. Canonical axis templates in a canonical frame (principal axis $\hat{\mathbf{z}}$). Each entry lists axis sets with notation (order:count). Lines are undirected (antipodal identified).

Family	Axis sets (order:count)
C_∞	$\{\pm\hat{\mathbf{z}}\}$ $(\infty:1)$
D_∞	$\{\pm\hat{\mathbf{z}}\}$ $(\infty:1)$ $\cup$ $\mathcal{R}_\infty$ $(2:\infty)$
C_n	$\{\pm\hat{\mathbf{z}}\}$ $(n:1)$
D_n	$\{\pm\hat{\mathbf{z}}\}$ $(n:1)$ $\cup$ $\mathcal{R}_n$ $(2:n)$
A_4	$\mathcal{T}_4$ $(3:4)$ $\cup$ $\mathcal{E}_3$ $(2:3)$
S_4	$\mathcal{X}_3$ $(4:3)$ $\cup$ $\mathcal{T}_4$ $(3:4)$ $\cup$ $\mathcal{B}_6$ $(2:6)$
A_5	$\mathcal{V}_6$ $(5:6)$ $\cup$ $\mathcal{F}_{10}$ $(3:10)$ $\cup$ $\mathcal{E}_{15}$ $(2:15)$

Symbols.

$$\hat{z} = (0,0,1),$$
$$\mathcal{R}_n = \{(\cos\tfrac{2\pi k}{n}, \sin\tfrac{2\pi k}{n}, 0)\}_{k=0}^{n-1},$$
$$\mathcal{R}_\infty = \{(\cos\theta, \sin\theta, 0) \mid \theta \in [0,\pi)\},$$
$$\mathcal{X}_3 = \{\pm e_x, \pm e_y, \pm e_z\}/\pm,$$
$$\mathcal{T}_4 = \tfrac{1}{\sqrt{3}}\{(s_1,s_2,s_3) \mid s_i \in \{\pm 1\},\ s_1 s_2 s_3 = 1\},$$
$$\mathcal{E}_3 = \{e_x, e_y, e_z\},$$
$$\mathcal{B}_6 = \tfrac{1}{\sqrt{2}}\{\text{perms of } (\pm 1, \pm 1, 0)\}/\pm,$$
$$\phi = \tfrac{1+\sqrt{5}}{2},$$
$$\mathcal{V}_6 = \tfrac{1}{\sqrt{1+\phi^2}}\{(0,\pm 1,\pm\phi),(\pm 1,\pm\phi,0),(\pm\phi,0,\pm 1)\}/\pm,$$
$$\mathcal{F}_{10} = \{\text{norm}(v_i+v_j+v_k) \mid (v_i,v_j,v_k) \in \text{faces}\}/\pm,$$
$$\mathcal{E}_{15} = \{\text{norm}(v_i - v_j) \mid (v_i,v_j) \in \text{edges}\}/\pm\,.$$

Each set defines canonical axis directions; $\pm$ denotes antipodal equivalence.

Table 10. Summary of family-level symmetry type confusions.

GT	Pred	Count	Datasets
C_∞	C_n	2	GSO(2)
C_n	C_∞	9	GSO(3), DSRSTO(2), BOP(4)
D_n	D_∞	1	GSO(1)
C_∞	D_∞	1	GSO(1)
C_n	D_n	14	GSO(8), DSRSTO(3), $\text{DSRSTO}_{\text{tex}}$(3)

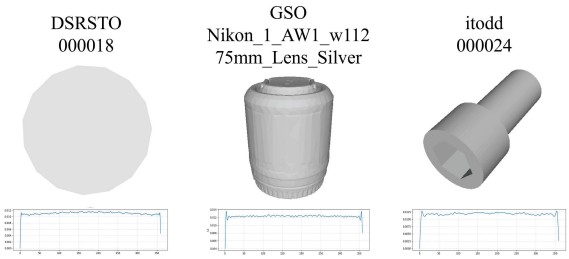

Figure 7. Representative objects exhibiting ambiguous rotational signatures. The examples show discrete shapes whose alignment-error profiles appear nearly continuous, with oscillations too weak or unstable to support reliable order inference.

9. Failure Modes

9.1. Type Misclassification (Family-Level Errors)

All family-level misclassifications are summarized in Table 10. They concentrate into two systematic patterns: discrete–continuous confusion and cyclic–dihedral confusion. These errors reflect structural limitations of the underlying geometric cues used for axis reasoning, rather than incidental prediction noise.

Discrete–Continuous Confusion. This failure mode appears in two characteristic situations. First, objects with very high discrete orders produce alignment-error curves whose oscillations are too weak to distinguish from a continuous profile. In our test sets, discrete symmetries appear up to 15-fold; orders up to 9 can be reliably identified, whereas 12-fold and 15-fold objects increasingly exhibit this near-continuous behavior and are partially interpreted as continuous. Second, globally continuous objects containing small discrete elements may generate only faint periodic cues that are easily suppressed during denoising. In both cases, the dominant-axis stage cannot recover a stable n-fold pattern and therefore classifies the symmetry as continuous. Representative objects and their alignment-error profiles are shown in Fig. 7.

A potential mitigation is to adopt a more frequency-sensitive order estimation, for example by analyzing the rotational signal in the spectral domain or using angular descriptors that amplify weak periodic components. Such strategies may improve sensitivity to subtle high-order oscillations while remaining robust to noise.

Figure 8. Representative cases of cyclic–dihedral confusion. Each object exhibits only mild differences between its two ends or weak prismatic components, making the orthogonal-axis test insufficiently discriminative between C-type and D-type families.

Table 11. Summary of order-level rotational symmetry confusions.

Type	GT	Pred	Count
C_n	2	12	1
C_n	2	9	1
C_n	2	4	3
C_n	6	3	2
D_n	4	2	5

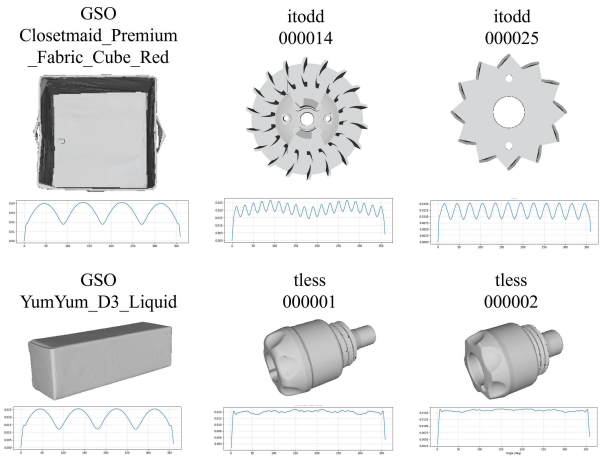

Figure 9. Representative examples of rotational order misestimation. Local symmetry-breaking features or sparse defects yield weak or mixed-frequency oscillations in the alignment curve, preventing reliable recovery of the true n-fold order.

Cyclic–Dihedral Confusion. These errors concentrate on objects whose two ends are not exactly identical but differ only weakly at the scale of the whole shape. Our family decision between C-type and D-type relies on the secondary-axis test: we search for an orthogonal axis and accept a dihedral family when the minimal alignment loss along the ring lies below an admissible threshold. For objects with slightly mismatched end caps or weak prismatic structures, the Chamfer loss remains low enough that the orthogonal axis is still accepted, leading the method to interpret a nearly cyclic object as dihedral. Representative examples are shown in Fig. 8.

A possible mitigation is to augment the secondary-axis decision with more global evidence along the main axis, for example by comparing cross-sectional profiles at multiple heights or by enforcing consistency of the prismatic pattern. Such cues could make the C/D distinction more robust when end-face asymmetries are subtle.

9.2. Order Misestimation (n-fold Errors with Correct Type)

Even when the symmetry family is correctly identified, the rotational order n may still be misestimated. Table 11 summarizes all cases in which the predicted order differs from the geometric ground truth, and Fig. 9 provides representative examples of these failures. The common pattern is that the periodic alignment signal becomes either too weak or distorted by local geometric irregularities, making nearby orders indistinguishable.

A key limitation is the reliance on a global Chamfer signal, which is insensitive to small but decisive symmetry-breaking structures. Local features such as handles, recesses, holes, or asymmetric trims are spatially averaged in the global alignment curve and therefore do not produce a clear n-fold signature. In nearly symmetric high-order shapes, sparse defects further generate mixed oscillations composed of weak low-frequency cues overlaid by dominant high-frequency components; the latter numerically dominate, causing the estimator to infer an incorrect order.

A potential improvement is to incorporate localized periodic evidence into the estimator. Radial motif detection or angular descriptors computed along local profiles could strengthen sensitivity to small order-breaking features, while spectral analysis of the alignment curve may help separate mixed frequencies and improve robustness for high-order objects.

9.3. Interpreting the Lower Accuracy on the C_n Family

The results in Table 2 show that objects labeled as C_n achieve noticeably lower accuracy than the other symmetry families. In addition to the methodological limitations discussed in Sections 9.1 and 9.2, this behavior follows from the structural role of C_n within the Geometric Degeneration Hierarchy (GDH) and from the heterogeneous nature of the objects assigned to this family.

Within the GDH, C_n corresponds to the weakest form of rotational structure, positioned only slightly above the non-symmetric case. Many objects fall into this category as degenerate forms of stronger symmetries: small disruptions such as asymmetric rims, recesses, attachments, cross-sectional irregularities, or texture-induced reductions readily break dihedral, continuous, or Platonic configurations and collapse them into a finite cyclic type. As a result, the C_n group contains a wide collection of borderline or partially broken shapes rather than a coherent family. These

Table 12. Comparison between our formulation and Wang et al.

Aspect	**Ours (KASALv2)**	**Wang et al.[26]**
Optimization objective	Differentiable Chamfer loss	Non-differentiable e_{ADI}
Search strategy	Two-stage sampling + gradient refinement	Dense spherical sampling + exhaustive search
Supported outputs	Type, order, full axis set	Axis localization only
Automation level	Fully automatic (no manual input)	Requires manual type/order per object

Table 13. Normalized alignment error e_{ADI}/d computed with and without excluding objects whose symmetry type or rotational order is misclassified.

Dataset	ours(KASALv2)		Wang et al.	
	Excl. wrong	All objects	Excl. wrong	All objects
DSRSTO	0.00212	0.00236	0.00172	0.00168
BOP	0.00256	0.00279	0.00290	0.00315
GSO	0.00262	0.00270	0.00256	0.00250

Table 14. Texture-induced degenerations permitted under the GDH. Each geometric family may reduce to one of the listed types when appearance lowers the effective rotational order.

Geometric Type	Admissible Texture-Induced Types
D_∞	$C_\infty,\ D_n,\ C_n$
D_n	$D_n,\ C_n$
C_∞	C_n
C_n	C_n
A_5	$A_4,\ C_n$
S_4	$A_4,\ C_n,\ D_n$
A_4	C_n

structural characteristics also make symmetry recovery difficult. Weak periodic cues, localized violations, and mixed-frequency oscillations often obscure the true discrete order and destabilize axis reasoning.

9.4. Summary and Future Work

Most failure cases arise from sensitivity limits in the geometric signals used by our estimator. The global Chamfer curve averages over small asymmetries, making local symmetry-breaking cues too weak to influence the objective, while the secondary-axis verification and periodicity test become ill-conditioned on near-continuous high-order patterns and weakly prismatic shapes. These issues reflect structural constraints of the current formulation.

Future work will reinforce the estimator by incorporating localized periodic cues, enforcing cross-sectional consistency to better separate prismatic and near-continuous cases, and analyzing rotational responses in the frequency domain to disentangle mixed components before order selection. These extensions should improve robustness on weak-symmetry objects, reduce mixed-frequency ambiguity, and stabilize order and family predictions in high-order regimes.

10. Additional Methodological and Experimental Details

10.1. Additional Evaluation on Localization Metrics

In addition to the results reported in the main paper, we also evaluate the localization accuracy e_{ADI}/d without excluding objects whose symmetry type or rotational order is misclassified. This removes any benefit from discarding our own failure cases and reflects a fully automatic end-to-end setting in which the predicted symmetry is used directly for axis localization. As shown in Table 13, the resulting e_{ADI}/d values remain close to those under the exclusion protocol, indicating that localization is stable even when upstream predictions are imperfect.

It is also important to note that our optimization does not use e_{ADI}/d as its objective: refinement relies entirely on a differentiable Chamfer-based loss to enable gradient-based learning in PyTorch. Despite this mismatch between the optimization target and the evaluation metric, our performance remains comparable to methods that explicitly minimize e_{ADI}, and the small numerical differences in the table should not be interpreted as a limitation of the method. For completeness, Table 12 summarizes the main differences between our formulation and that of Wang et al.[26], including objective functions, search strategies, supported outputs, and automation level.

10.2. Texture-Symmetry Implementation Details

Appearance-induced symmetry is estimated as an extension of the geometric analysis described in the main paper. We begin by recovering all geometric symmetry axes of the object using the standard KASALv2 pipeline. The point cloud is then augmented with color information by pairing each (x, y, z) coordinate with its corresponding (r, g, b) value so that texture and geometry are processed in a unified manner. Once the geometric axes are obtained, appearance periodicity is evaluated by rotating the colored point cloud through 360° about each axis and computing the Chamfer distance between the original cloud and its rotated copies. This produces a one-dimensional signal that reflects the rotational

42	2023-08-22	ZeroPose-One-Hypo	RGB-D	0.348	0.272	0.156	0.536	0.307	0.362	0.462	0.341	9.756
43	2026-01-29	Foundationpose (with sym piror)	RGB-D		0.720	0.601		0.653	0.664		0.879	
44	2026-01-29	Foundationpose (without sym piror)	RGB-D		0.718	0.593		0.644	0.652		0.878	
45	2024-07-20	GZS6D-BP(coarse+refine)	RGB-D		0.671		0.907	0.544	0.584	0.795	0.758	

Figure 10. Official BOP leaderboard results for FoundationPose trained with and without the rotational-symmetry prior. Both models are trained from scratch on a synthetic GSO training set under identical hyperparameters. The version augmented with symmetry priors achieves slightly higher accuracy across the benchmark.

repeatability of the texture. The number of oscillations in this signal is taken as the texture-induced order associated with that axis, yielding a complete set of appearance orders aligned with the underlying geometric structure.

These appearance orders are then compared with the axis configurations permitted under the GDH for each family into which the object may degenerate. Both axis count and discrete orders must satisfy the admissible degeneration patterns. Table 14 summarizes all allowed texture-driven degenerations, and Fig. 11 illustrates a representative example of this procedure.

10.3. FoundationPose Training Settings and Public BOP Results

Section 4.4 describes how KASALv2-derived symmetry priors are used in FoundationPose training. Here we provide the implementation details.

For each ground-truth pose of a rotationally symmetric object, we construct its symmetry-equivalent rotation set from the symmetry type, rotational order, and axis predicted by KASALv2. For finite discrete symmetries, the set is obtained by direct enumeration. For continuous symmetries, the symmetry space is discretized into a finite set of equivalent rotations following the standard BOP protocol. We then select the rotation closest to a fixed reference rotation as the canonical training label. In our implementation, the identity rotation is used as the default reference, although any fixed reference rotation would serve the same role.

We train the RefineNet of FoundationPose from scratch on a synthetic GSO training set of about one million rendered images. Training is conducted on an NVIDIA 4090 GPU with a learning rate of 0.0001, the Adam optimizer, and a batch size of 128. The loss function follows the original FoundationPose design, and training runs for 20k iterations. During training, FoundationPose learns to refine candidate poses whose angular deviations from the ground-truth poses are within 20°.

During testing, 41 vertices are sampled using the Fibonacci algorithm. For each vertex, six in-plane rotation angles are applied, resulting in 246 candidate poses in total. Each candidate pose is refined independently, and the optimal pose is selected using FoundationPose's ScoreNet. The 2D detections used for testing are provided by MUSE.

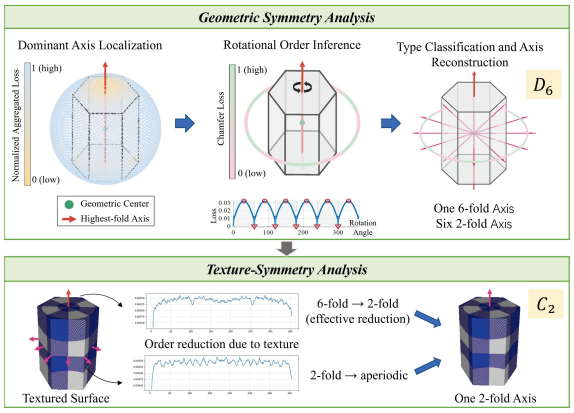

Figure 11. Example of texture-induced order reduction. Top: geometric symmetry analysis recovers a D_6 structure with one 6-fold axis and six 2-fold axes. Bottom: rotating the textured surface around each geometric axis shows that texture breaks all 2-fold axes (no periodicity) and reduces the 6-fold axis to 2-fold, yielding a final texture-aware classification of C_2.

The top-k 2D bounding boxes are selected according to the MUSE confidence scores, where k is determined by the number of target objects of each category in the BOP 6D Localization task for each image. FoundationPose is then applied only to these top-k bounding boxes for 6D pose estimation.

To verify that the improvement is not checkpoint-specific, we additionally report a retrained multi-checkpoint evaluation on five BOP datasets. The reported numbers are averaged over five checkpoints from the same training run (seed 0). The resulting pose estimates are submitted to the BOP website, and screenshots of the public results are shown in Fig. 10. These results are publicly available on the BOP website.

10.4. Hardware, Runtime, and Open-Source Availability

All timing measurements refer only to the KASALv2 symmetry analysis pipeline. On an Intel Core i5-13400F CPU and an NVIDIA RTX 3050 (6 GB), the full pipeline, including dominant-axis detection, order estimation, type classification, and full-axis reconstruction, runs in about 1.2–

1.7 s per object, depending on object type and noise level. Chamfer evaluation and local refinement are implemented in PyTorch and executed on the GPU. KASALv2 serves as an offline symmetry annotator that processes objects independently and can be parallelized across GPUs or machines. The current implementation relies on standard PyTorch operators; optimized Chamfer modules or higher-end hardware could further reduce runtime without changing the algorithm. The complete KASALv2 codebase, together with the re-annotated BOP symmetry labels, is available to support reproducibility and downstream research requiring high-quality symmetry annotations.