Shape Abstraction via Marching Differentiable Support Functions

Supplementary Material

8. Differentiable Contact Features

In [4], the DSF is adjusted by the pose q inducing rotation $R(q)$ and translation $t(q)$:

$$\bar{h}(q,x) = h(R(q)^T x) + t(q)^T x \tag{18}$$

$$\bar{s}(q,x) = R(q)s(R(q)^T x) + t(q) \tag{19}$$

where $s(x) = \frac{\partial h}{\partial x}$ is a support point. Then, the objective function for collision detection is defined as

$$F(x,\sigma,q) = \begin{bmatrix} \sigma(\bar{s}_i - \bar{s}_j) + (1-\sigma)(t_i - t_j) \\ \|x\|^2 \end{bmatrix} \tag{20}$$

where $\bar{s}_i = R_i s_i(R_i^T x) + t_i$ and $\bar{s}_j = R_j s_j(-R_j^T x) + t_j$ are derived from Eq. (19) for each pose q_i and q_j, and $\sigma \in \mathbb{R}$ is a scaling factor for both DSF to *touch* each other. The nonlinear equation $F(x,\sigma,q) = 0$ is solved by the contact feature solver in [4].

On the other hand, to compute the contact features based on our shape parameters ϕ, Eq. (20) is reformulated as

$$F(x,\sigma,\phi_i,\phi_j) = \begin{bmatrix} \sigma(s_i - s_j) + (c_i - c_j) \\ \|x\|^2 \end{bmatrix} = 0 \tag{21}$$

where $s_i = s(x,\phi_i)$ and $s_j = s(-x,\phi_j)$ are the support points for each DSF, and c is the center of each DSF. Then, the Jacobian of F is computed as follows:

$$J = \left[\frac{\partial F}{\partial x}, \frac{\partial F}{\partial \sigma}\right] = \begin{bmatrix} \sigma\left(\frac{ds_i}{dx} + \frac{ds_j}{dx}\right) & s_i - s_j \\ 2x^T & 0 \end{bmatrix}. \tag{22}$$

Since Eq. (21) and Eq. (22) are special cases of Eq. (20) and its Jacobian where $R = I$ and $t = c$, the same contact feature solver [4] can be applied in our study. Additionally, the derivatives of the contact features with respect to ϕ are computed using $\frac{dx^*}{d\phi}$ derived from the implicit function theorem as follows:

$$\frac{\partial F^*}{\partial \phi} + J^* \left[\frac{dx^*}{d\phi}; \frac{d\sigma^*}{d\phi}\right] = 0 \tag{23}$$

where F^* and J^* are Eq. (21) and Eq. (22) evaluated at the solution (x^*, σ^*) of the nonlinear equation $F(x,\sigma,\phi_i,\phi_j) = 0$. Similarly, the differentiable contact features with respect to pose, represented by rotation I and translation c, can also be computed using the implicit function theorem.

9. Metrics

In Sec. 5, two metrics are used for evaluation: intersection over union (IoU) and Chamfer-L1 distance. The IoU is defined in terms of volumes as

$$\text{IoU} = \frac{\text{Vol}(A \cap B)}{\text{Vol}(A \cup B)} \tag{24}$$

where A and B are the geometries, and $\text{Vol}(A)$ is the volume of A, approximated using 10^5 points uniformly sampled from the input mesh. The IoU is computed based on the number of point samples within each geometry. Specifically, $\text{Vol}(A \cap B)$ is the number of samples inside of both A and B, and $\text{Vol}(A \cup B)$ is the number of samples inside either A or B. To compute these volumes, the implicit function representing the superquadrics, and the minimum distance solver for DSF [4] are used for each primitive.

The Chamfer-L1 distance is defined as

$$\begin{aligned} d(A,B) = &\frac{1}{|A_s|} \sum_{p \in A_s} \min_{q \in B_s} \|p - q\|_1 + \\ &\frac{1}{|B_s|} \sum_{q \in B_s} \min_{p \in A_s} \|p - q\|_1 \end{aligned} \tag{25}$$

where A_s and B_s are the surface points sampled from geometries A and B, respectively. The equal-distance sampling for superquadrics [5] is determined by the distance parameter, leading to variations in sample sizes among different objects. For comparison, the surface points from DSF are downsampled to match the number of samples from superquadrics, which ranges from 28K to 50K.

10. Implementation Details

For our method, the SDF data P, S, and G are sampled from voxelized SDF with a resolution of 100^3. Each data contains 100K samples, with 50K data sampled from the voxels inside of the object and 50K data sampled from the voxels near the surface of the object. In Alg. 3, the stopping threshold t_{end} is set to -0.5 times the voxel grid interval, and the method-conversion threshold t_s is set to -2 times the voxel grid interval. For self-degeneration in Sec. 4.4, a fitted DSF is removed if it satisfies the following condition:

$$\frac{N_{out}}{N_{in}} > 0.3 \quad or \quad N_{DSF} < 20 \tag{26}$$

where N_{DSF} is the number of points inside of DSF, N_{in} is the number of points inside of DSF with SDF value $s \leq 0$, and N_{out} is the number of points inside of DSF with SDF value $s > 0$.

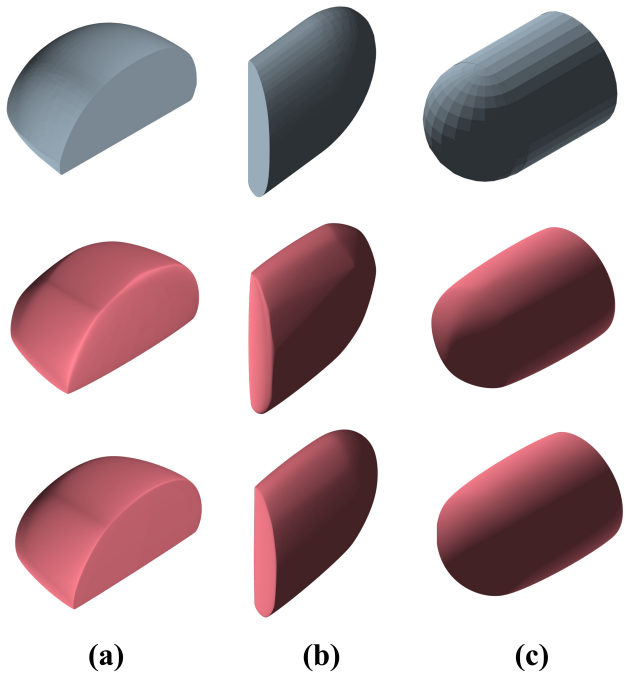

Figure 12. DSF fitting results using 20 vertices with α (bottom) and without α (middle, where $\forall\alpha_i$ is fixed at 0 for geometric center) for the given convex meshes (top).

DSF Fitting Cost	(a)	(b)	(c)
Without α	0.0114	0.0061	0.0233
With α	0.0054	0.0023	0.0055

Table 3. The resulting costs for the DSF fitting computed by the objective function Eq. (5). The convex meshes (a-c) are visualized in Fig. 12.

11. The Center Parameters of DSF

The effect of the center parameters α on fitting results is discussed in Sec. 3.1. In this section, we provide additional visualizations of fitting results for 3D shapes with and without α. In Fig. 12, DSFs are fitted to convex meshes with both curved and flat surfaces. The resulting costs computed using Eq. (5) are summarized in Tab. 3 for the three meshes in Fig. 12. Involving α in the shape parameters ϕ enhances the representation performance of sharp edges between curved and flat surfaces. While the fitting cost depends on object geometry and scale, involving α consistently yields lower costs compared to fixed α. This improvement becomes more significant when the DSF has fewer vertices.

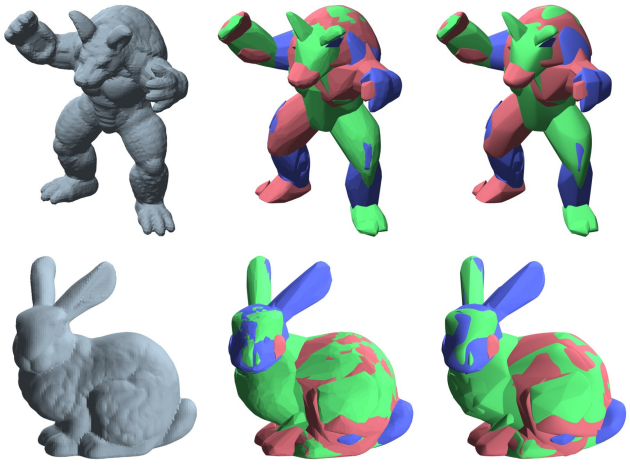

Figure 13. DSF abstraction results. The object is the armadillo and the Stanford rabbit from the Stanford 3D scanning repository [2]. For each input object (left), the hyperplane samples for the DSF fitting are obtained from each convex region R generated within the object (middle, converted into the polygons). Then, the DSFs are fitted to each hyperplane set (right).

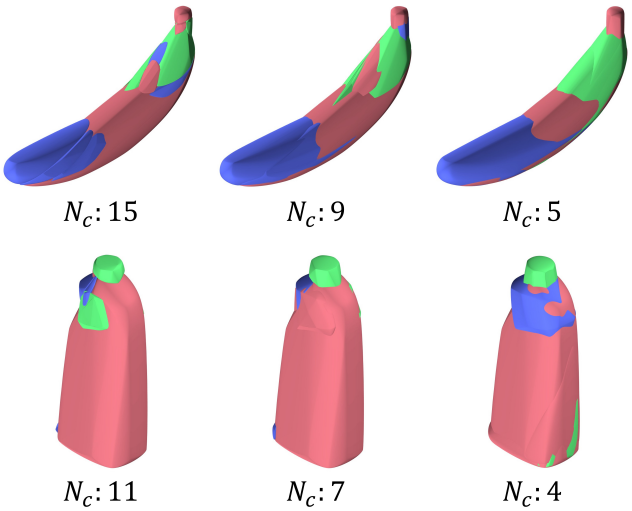

Figure 14. DSF abstraction results using the region reduction method (Alg. 1) with overlap parameter $o_d = 0$ (left), $o_d = 0.012$ (middle), and $o_d = 0.6$ (right) for the normalized 3D space. The number of DSFs (N_c) is presented below each result.

12. Visualizations of Hyperplane Samples in Abstraction Process

In this section, the visualizations of the DSF fitting results in Alg. 3 are presented. Each set of hyperplanes used for the DSF fitting Eq. (5) can be converted into a polygon. In Fig. 13, these polygons are visualized with the resulting abstracted DSFs. Despite the sharp surfaces of the polygons, the DSF fitting process successfully reconstructs the smoothness of the geometries.

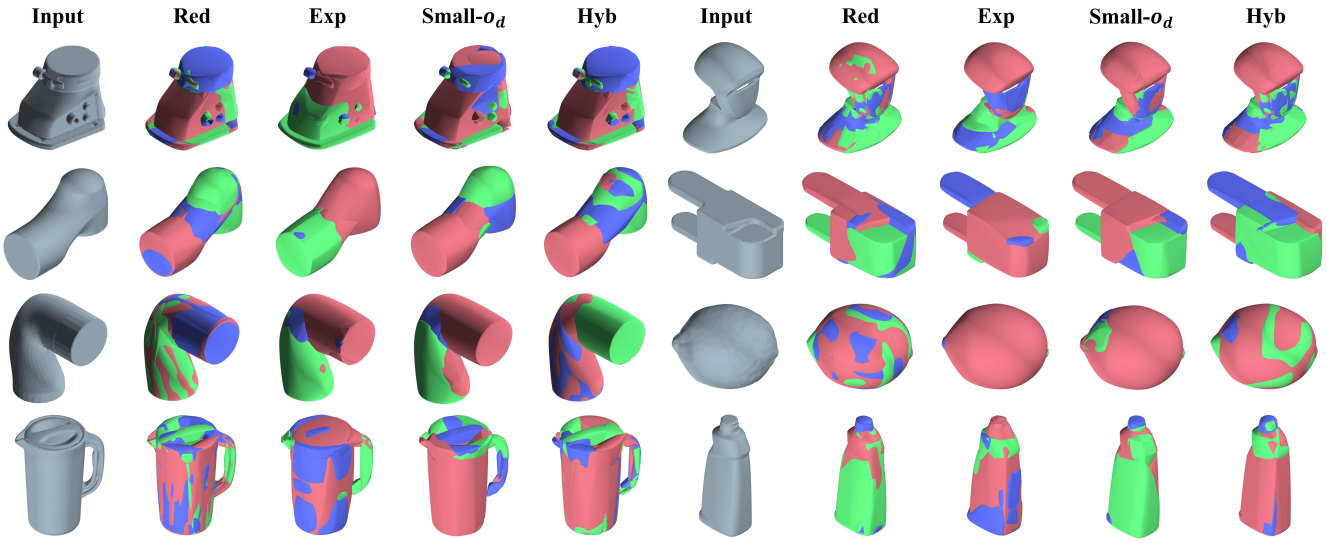

Figure 15. Abstraction results for the robot links [8] and the YCB dataset [1]. From left to right: Input mesh, the region reduction method (Alg. 1, Red), the region expansion method (Alg. 2, Exp), the hybrid method with a small o_d (Alg. 3, Small-o_d), and the hybrid method with a large o_d (Hyb). The results of Hyb is the same as MDSF in Sec. 5.

	IoU ↑				Chamfer-L1 ↓				N_c ↓			
Category	Red	Exp	Small-o_d	Hyb	Red	Exp	Small-o_d	Hyb	Red	Exp	Small-o_d	Hyb
Franka-fr3	0.9537	0.9425	**0.9615**	0.9517	0.0823	0.0961	**0.0799**	0.0847	16.875	**9**	20.375	13
Lite6	0.9518	0.9386	**0.9593**	0.9533	0.0935	0.1079	**0.0869**	0.0919	7.8571	7.5714	14.2857	**7.2857**
Xarm7	0.9533	0.9468	**0.9578**	0.9535	0.0857	0.0960	**0.0852**	0.0863	18.875	**9.375**	21.75	14.25
A1	0.9399	0.9259	**0.9459**	0.9424	0.0651	0.0753	**0.0627**	0.0650	10.8	**7.4**	11.8	9.8
G1	0.9331	0.9096	**0.9349**	0.9276	**0.0781**	0.0952	0.0786	0.0809	18.5652	**15.1304**	20	15.2174
YCB	**0.9296**	0.8834	0.9137	0.9076	**0.0677**	0.0883	0.0741	0.0764	22	**14.4286**	21.9286	16.5714
mean	**0.9399**	0.9170	**0.9399**	0.9333	**0.0780**	0.0938	0.0783	0.0810	17.3846	**12.1077**	19.4308	13.8462

Table 4. Results of the ablation studies. Red: the region reduction method (Alg. 1). Exp: the region expansion method (Alg. 2). Small-o_d: the hybrid method with a small o_d (Alg. 3). Hyb: the hybrid method with a large o_d. Chamfer-L1 is multiplied by 10. The number of objects for each category is 8, 7, 8, 5, 23, and 14, from top to bottom. The results of Hyb is the same as MDSF in Sec. 5.

13. Overlap Parameters

The abstraction results for different overlap parameters o_d using the region reduction method (Alg. 1) are visualized in Fig. 14. When $o_d = 0$, discontinuous gaps appear between the DSFs. For a small positive value of o_d (0.012 for the normalized 3D space), continuous surfaces are produced with fewer DSFs than $o_d = 0$. For a large o_d (0.6 for the normalized 3D space), the number of DSFs is smaller than $o_d = 0.012$, as the intersection between DSFs is perfectly allowed.

14. Results of Ablation Studies

In Sec. 5.2, the abstraction performances and the number of resulting DSFs are compared for each abstraction method: the region reduction method Alg. 1, the region expansion method Alg. 2, and the hybrid method Alg. 3 with the two o_d. These results are visualized in Fig. 15 for the robot links CAD data provided by the MuJoCo Menagerie [8] and the YCB dataset [1]. The IoU, Chamfer-L1 distance, and N_c for all categories are summarized in Tab. 4. The same hyperparameters as Sec. 5.2 are used for each method.

15. Additional Results

To supplement the limited visualizations of abstraction results in Fig. 8, additional results for all categories are provided in Figs. 16 to 21 using the same baselines as Sec. 5.1. The hyperparameters are consistent with those in Sec. 5.1, including the batch size and the maximum training steps for CvxNet [3], N_v and N_h for CoDSF and our method (MDSF), and the overlap parameter o_d for Alg. 1.

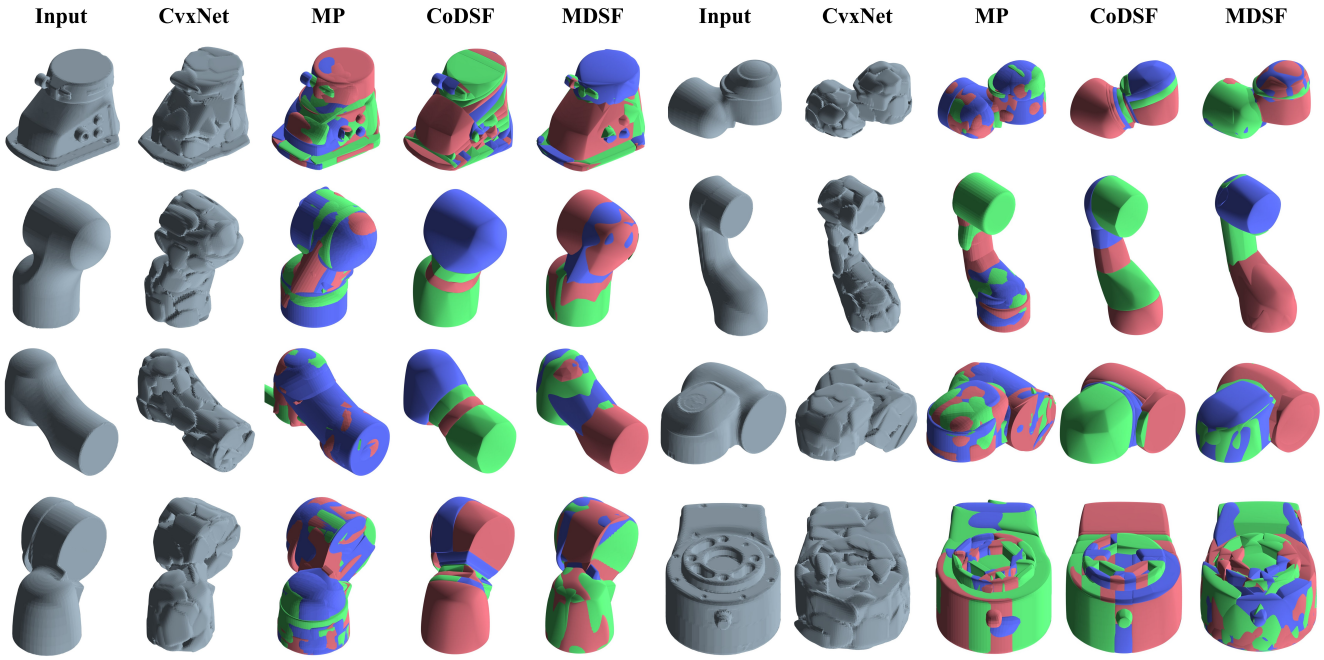

Figure 16. Abstraction results for the Franka-fr3 in the MuJoCo Menagerie [8]. From left to right: Input mesh, CvxNet [3], Marching-Primitives (MP [6]), DSF fitting for CoACD [7] results (CoDSF), and our DSF abstraction algorithm (MDSF, Alg. 3).

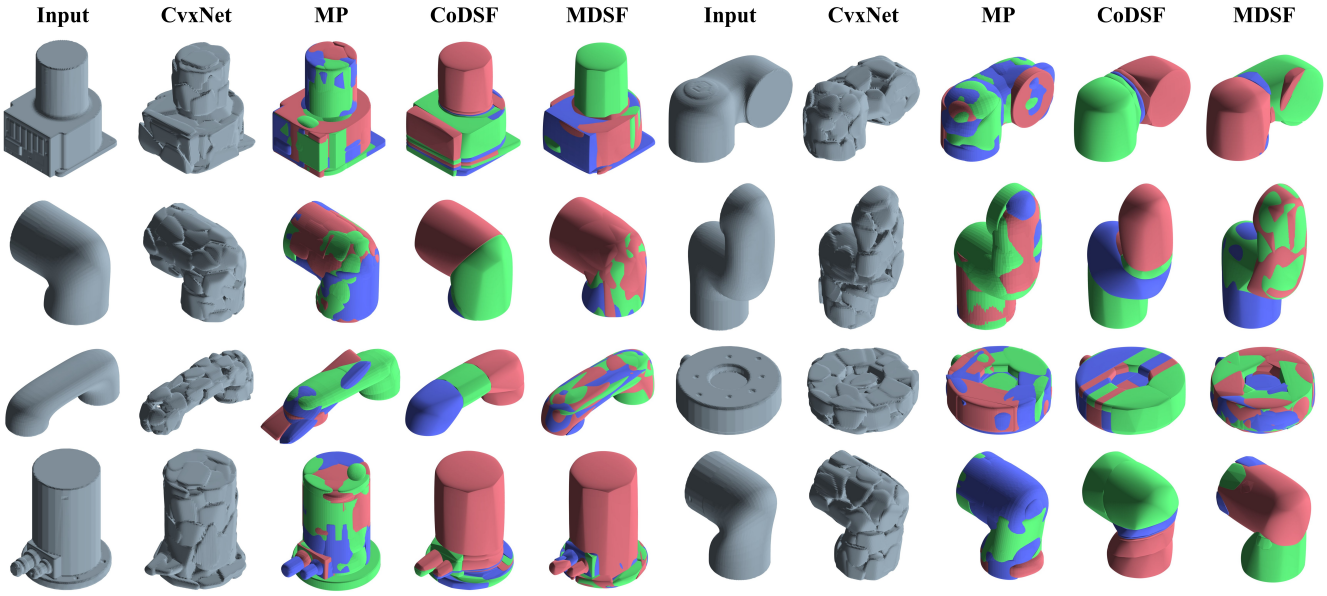

Figure 17. Abstraction results for the Lite6 (top three rows) and the Xarm7 (bottom row) in the MuJoCo Menagerie [8]. From left to right: Input mesh, CvxNet [3], Marching-Primitives (MP [6]), DSF fitting for CoACD [7] results (CoDSF), and our DSF abstraction algorithm (MDSF, Alg. 3).

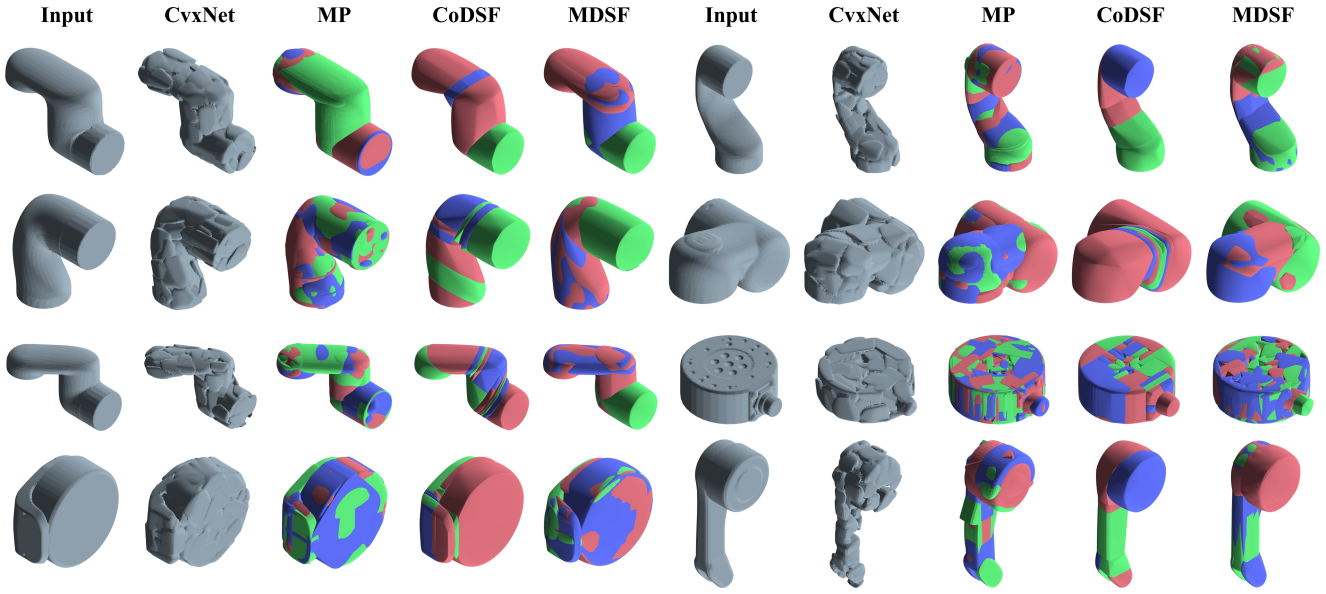

Figure 18. Abstraction results for the Xarm7 (top three rows) and the A1 (bottom row) in the MuJoCo Menagerie [8]. From left to right: Input mesh, CvxNet [3], Marching-Primitives (MP [6]), DSF fitting for CoACD [7] results (CoDSF), and our DSF abstraction algorithm (MDSF, Alg. 3).

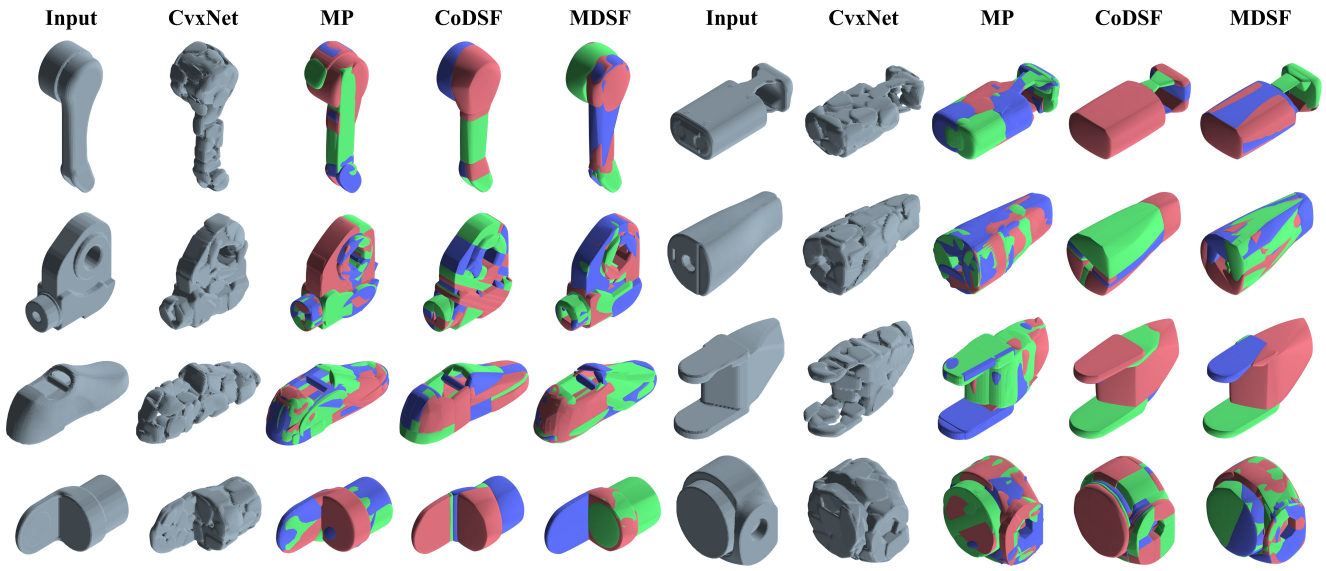

Figure 19. Abstraction results for the A1 (top row) and the G1 (bottom three rows) in the MuJoCo Menagerie [8]. From left to right: Input mesh, CvxNet [3], Marching-Primitives (MP [6]), DSF fitting for CoACD [7] results (CoDSF), and our DSF abstraction algorithm (MDSF, Alg. 3).

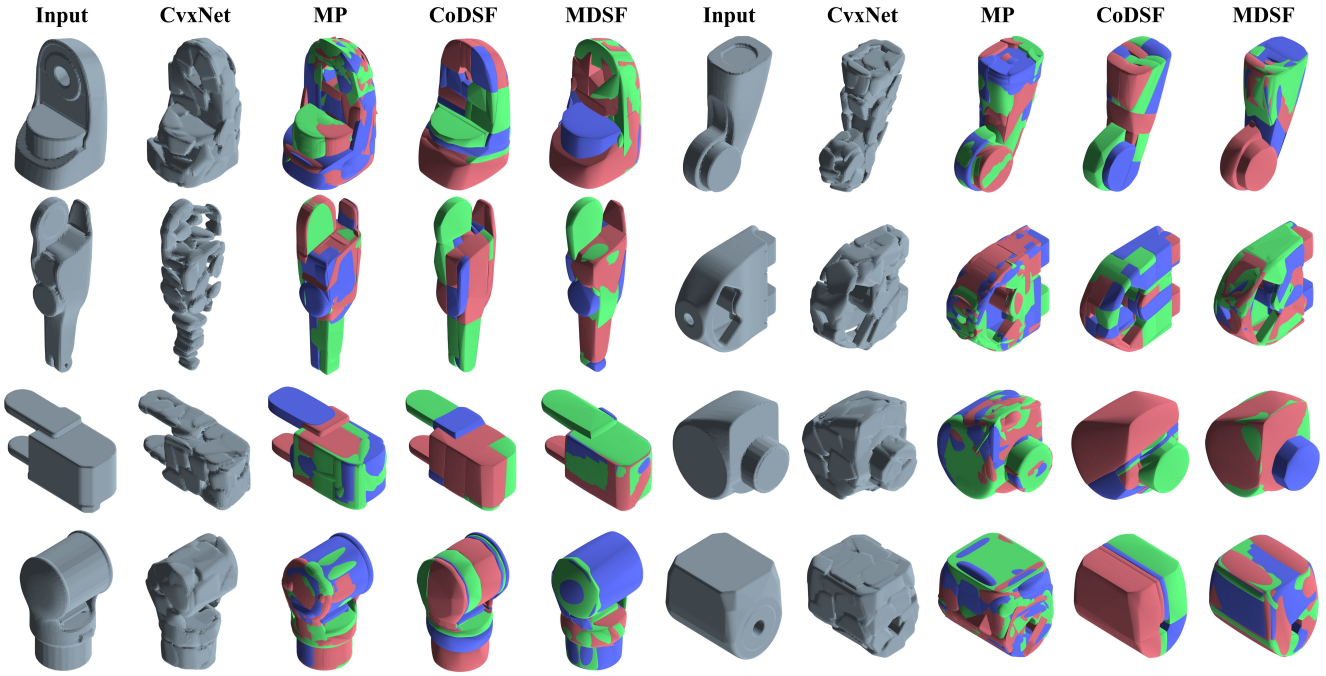

Figure 20. Abstraction results for the G1 in the MuJoCo Menagerie [8]. From left to right: Input mesh, CvxNet [3], Marching-Primitives (MP [6]), DSF fitting for CoACD [7] results (CoDSF), and our DSF abstraction algorithm (MDSF, Alg. 3).

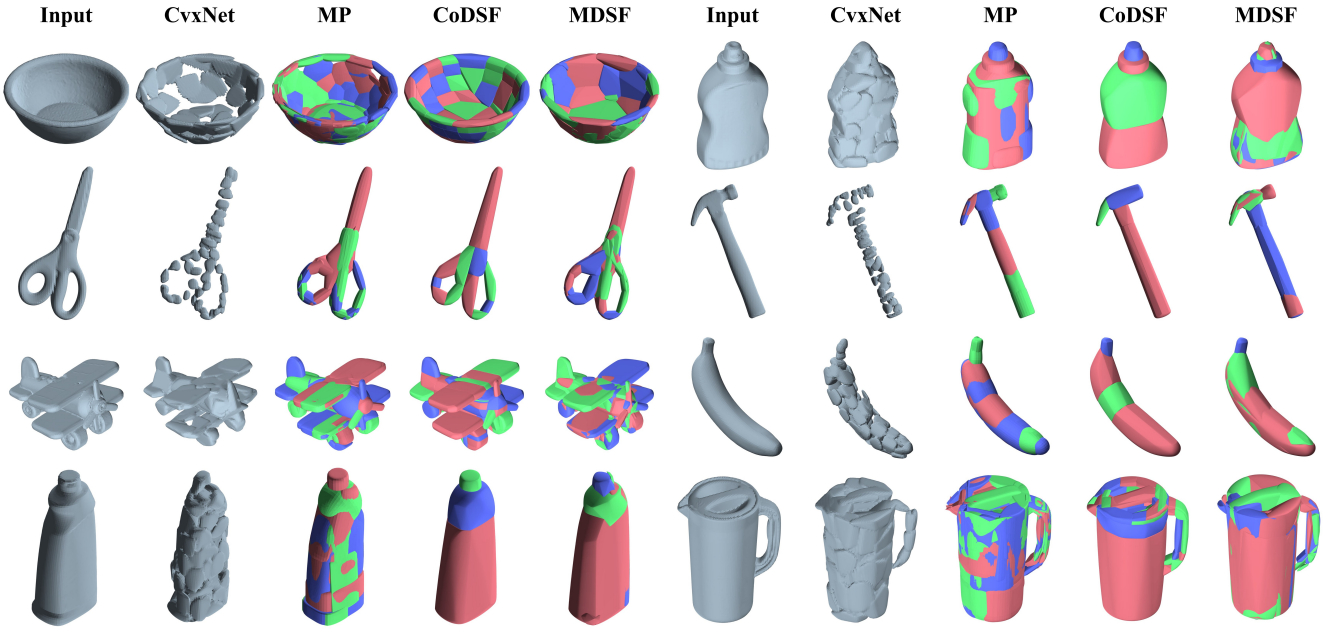

Figure 21. Abstraction results for the YCB dataset [1]. From left to right: Input mesh, CvxNet [3], Marching-Primitives (MP [6]), DSF fitting for CoACD [7] results (CoDSF), and our DSF abstraction algorithm (MDSF, Alg. 3).

References

[1] B. Calli, A. Singh, A. Walsman, S. Srinivasa, P. Abbeel, and A. M. Dollar. The ycb object and model set: Towards common benchmarks for manipulation research. In *2015 International Conference on Advanced Robotics (ICAR)*, pages 510–517. IEEE, 2015. 3, 6

[2] B. Curless and M. Levoy. A volumetric method for building complex models from range images. In *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*, pages 303–312, 1996. 2

[3] B. Deng, K. Genova, S. Yazdani, S. Bouaziz, G. Hinton, and A. Tagliasacchi. Cvxnet: Learnable convex decomposition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 31–44, 2020. 3, 4, 5, 6

[4] J. Lee, M. Lee, and D. Lee. Uncertain pose estimation during contact tasks using differentiable contact features. In *Proceedings of Robotics: Science and Systems*, 2023. 1

[5] W. Liu, Y. Wu, S. Ruan, and G. S. Chirikjian. Robust and accurate superquadric recovery: A probabilistic approach. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2676–2685, 2022. 1

[6] W. Liu, Y. Wu, S. Ruan, and G. S. Chirikjian. Marching-primitives: Shape abstraction from signed distance function. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8771–8780, 2023. 4, 5, 6

[7] X. Wei, M. Liu, Z. Ling, and H. Su. Approximate convex decomposition for 3d meshes with collision-aware concavity and tree search. *ACM Transactions on Graphics (TOG)*, 41 (4):1–18, 2022. 4, 5, 6

[8] K. Zakka, Y. Tassa, and MuJoCo Menagerie Contributors. MuJoCo Menagerie: A collection of high-quality simulation models for MuJoCo, 2022. 3, 4, 5, 6