Supplementary Material - DProST: Dynamic Projective Spatial Transformer Network for 6D Pose Estimation

Jaewoo Park[1,2] and Nam Ik Cho[1,2,3]

[1] Department of ECE & INMC, Seoul National University, Seoul, Korea
[2] SNU-LG AI Research Center, Seoul, Korea
[3] IPAI, Seoul National University, Seoul, Korea
{bjw0611,nicho}@snu.ac.kr

A Details of Pose Estimator

The disentangled representation is used for the pose estimator output as in [3]. For each iteration i, our ResNet34 [2] based pose estimator predicts relative translation on image space, which can be written as follows:

$$v_x^i = f_x \left(\frac{t_x^i}{t_z^i} - \frac{t_x^{i-1}}{t_z^{i-1}} \right), \tag{1}$$

$$v_y^i = f_y \left(\frac{t_y^i}{t_z^i} - \frac{t_y^{i-1}}{t_z^{i-1}} \right), \tag{2}$$

$$v_z^i = \frac{t_z^i}{t_z^{i-1}}, \tag{3}$$

where v_x^i and v_y^i are pixel-wise translation estimation, v_z^i is the relative scale of the object, f_x and f_y are focal lengths of x and y axis in intrinsic matrix, respectively, and t_x^i, t_y^i, and t_z^i are the components of the translation vector t_i. With the Equations (1), (2), and (3), we update the translation vector t_{i-1} to t_i for each iteration.

The network also predicts the two three-dimensional vectors e_1^i and e_2^i for rotation representation as in [3, 6], which can be converted to relative rotation matrix as follows:

$$r_1^i = \frac{e_1^i}{\|e_1^i\|_2}, \tag{4}$$

$$r_3^i = \frac{r_1^i \wedge e_2^i}{\|e_2^i\|_2}, \tag{5}$$

$$r_2^i = r_3^i \wedge r_1^i, \tag{6}$$

where $\wedge$ represents the cross product, and r_1^i, r_2^i, and r_3^i are the vectors in the relative rotation matrix of the i-th iteration.

To localize the initial projection of object in bounding box (x, y, w, h) as in [3], we set the initial translation t_0 as follows:

$$t_x^0 = (c_x - p_x)\frac{t_z^0}{f_x} \tag{7}$$

$$t_y^0 = (c_y - p_y)\frac{t_z^0}{f_y} \tag{8}$$

$$t_z^0 = \frac{d}{2}(\frac{f_x}{w} + \frac{f_y}{h}) \tag{9}$$

where t_x^0, t_y^0, and t_z^0 are components of initial translation vector, c_x and c_y are center of the bounding box, p_x and p_y are principal point, and d represents the diameter of object which is set to two in our normalized object setting. Additionally, we use the identity matrix for the initial rotation matrix R_0.

B Additional Ablation Studies

The performances of grid matching (GM), point matching (PM), and image matching (IM) objective function are visualized in Fig. B.1. The mean distance between object vertices in camera space is used for PM as in [3, 4], and mean-squared-error between projection in image space is used for IM as [1]. We confirm that the GM predict the rotation more accurately than PM in long objects such as glue in the LM dataset, and the shape bias in PM causes performance degradation. In detail, since the PM uses the object vertices, even if the actual rotation error of two predictions are same, the loss vary depending on the direction of misalignment. For example, the PM loss of the axial rotation error of a long object is relatively lower than other direction errors, which hinders accurate prediction of axial rotation. On the other hand, since the GM uses a uniformly distributed grid, no performance degradation due to shape bias has occurred. Furthermore, since the GM reflects the projective geometry on the grid shape and leverages it to predict the distance to object, the GM shows better results than PM in z-axis translation. Unlike the GM and PM, since the IM does not leverage the 3D location information, IM based model fails to predict the pose in some objects and shows low performance.

C Additional Qualitative Results

We demonstrate the additional example result of each iteration for the LMO and YCBV dataset in Fig. C.2 and Fig. C.4, respectively. In addition, we illustrate additional qualitative results of the LMO dataset and YCBV dataset in Fig. C.3 and Fig. C.5, respectively. Additionally, to demonstrate the competence of our method, we compare the qualitative results of our method and the other state-of-the-art methods in C.6. We also visualize reference feature quality in C.7 with the number of references. As shown in the figure, the more the reference view is used, the more accurate the shape becomes, but there is a trade-off in which the texture is blurred.

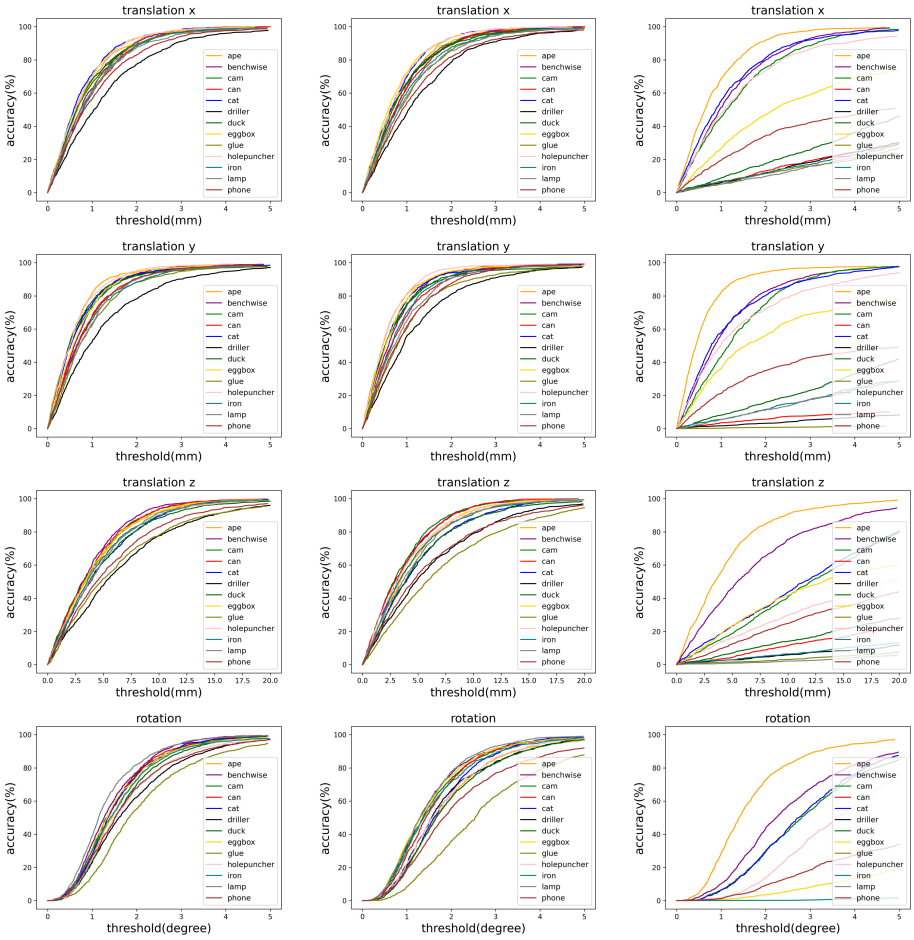

Fig. B.1. Comparison of accuracies according to loss functions on the LM dataset: Each column demonstrates the accuracy of GM loss, PM loss, and IM loss on the LM dataset, respectively. We visualize the translation error for each axis in the first three rows and rotation error in the last row.

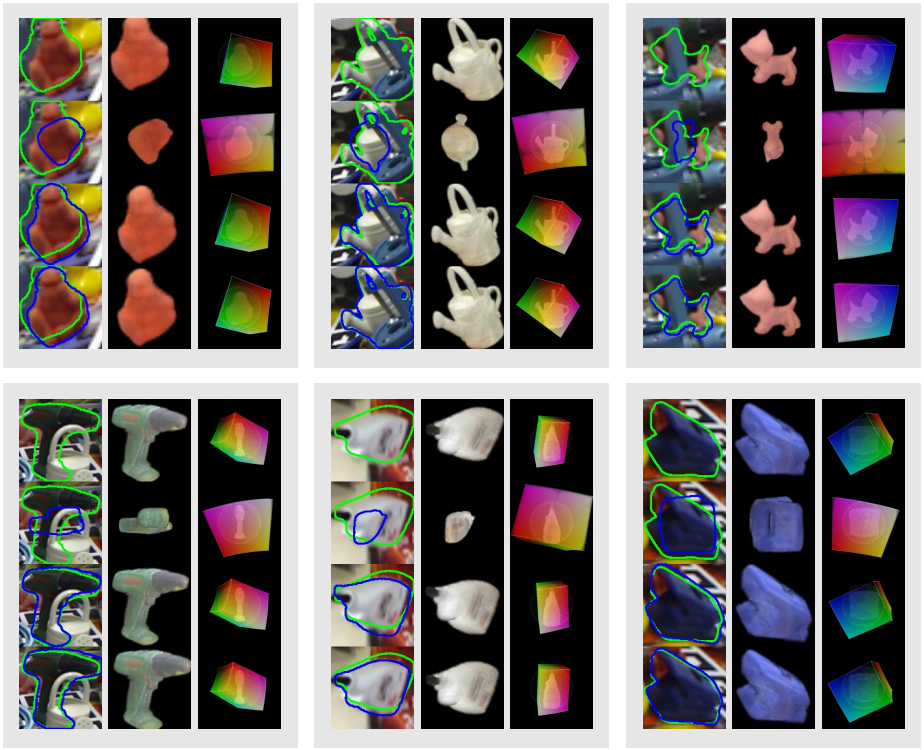

Fig. C.2. Qualitative results of iteration on the LMO dataset: For each column in the grey cell, we visualize the contours of the projection, projection by estimated pose, and object space grid. From the first row to the fourth row in each gray cell, the ground-truth pose, the initial pose, the first iteration pose, and the second iteration pose are visualized. The predicted and ground-truth poses are represented by blue and green contour, respectively.

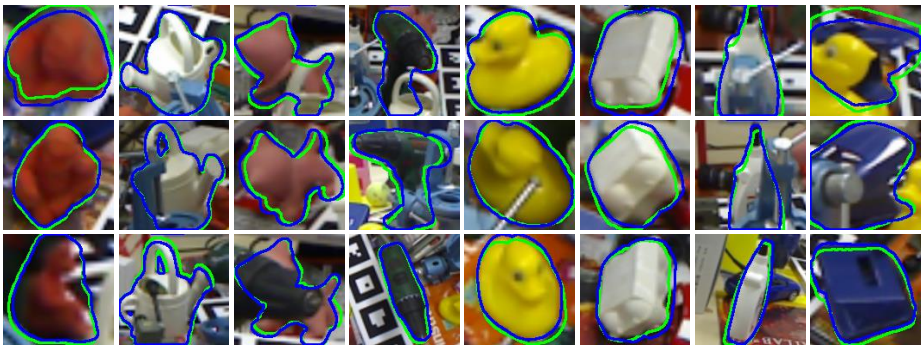

Fig. C.3. Qualitative results on the LMO dataset: We visualize additional results on the LMO dataset. The predicted pose's contour is represented by blue, and the ground truth pose's contour is demonstrated by green.

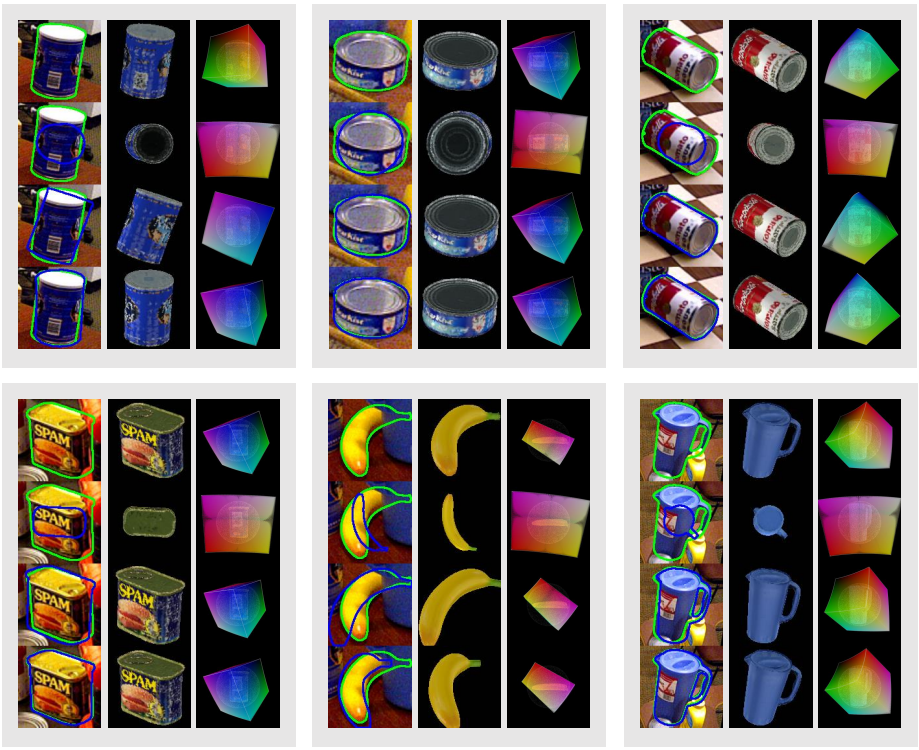

Fig. C.4. Qualitative results of iteration on the YCBV dataset: For each column in grey cells, we visualize the contours of the projection, projection results, and object space grid. From the first row to the fourth row in each gray cell, the ground-truth pose, the initial pose, the first iteration pose, and the second iteration pose are visualized, respectively. The predicted pose's contour is represented by blue, and the ground truth pose's contour is demonstrated by green.

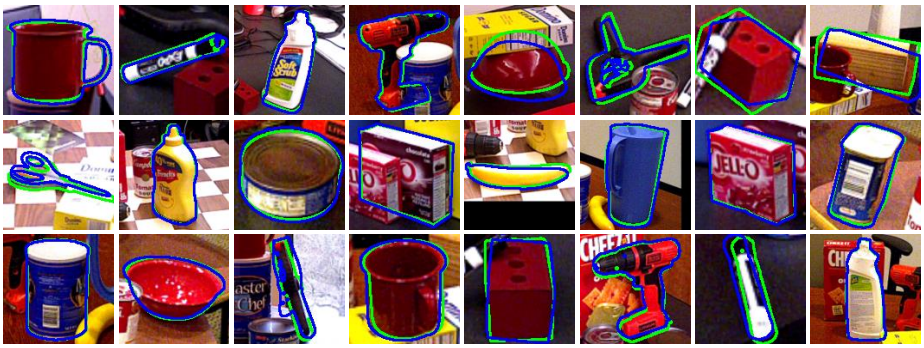

Fig. C.5. Qualitative results on the YCBV dataset: We visualize additional results on the YCBV dataset. The predicted pose's contour is represented by blue, and the ground truth pose's contour is demonstrated by green.

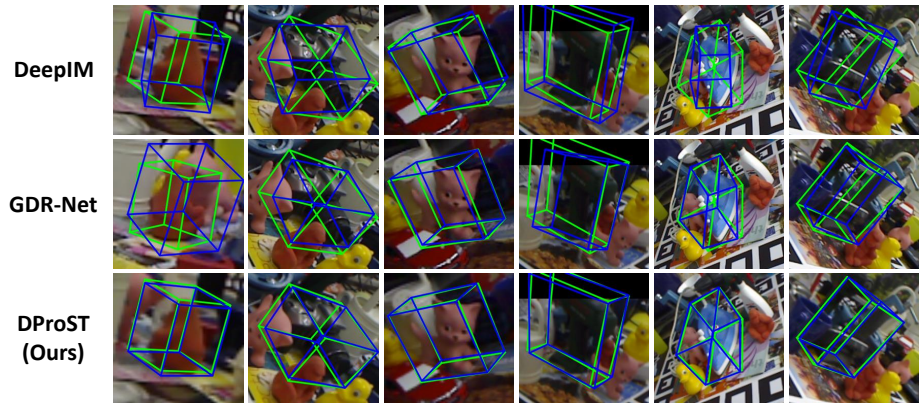

Fig. C.6. Comparison of qualitative results with other SOTA methods: For each row, we visualize the qualitative results of DeepIM [4] (first row), GDR-Net [5] (second row), and our method (last row) on the LM dataset. The green and blue boxes visualize the projection of the object 3D bounding box using ground-truth and predicted pose, respectively.

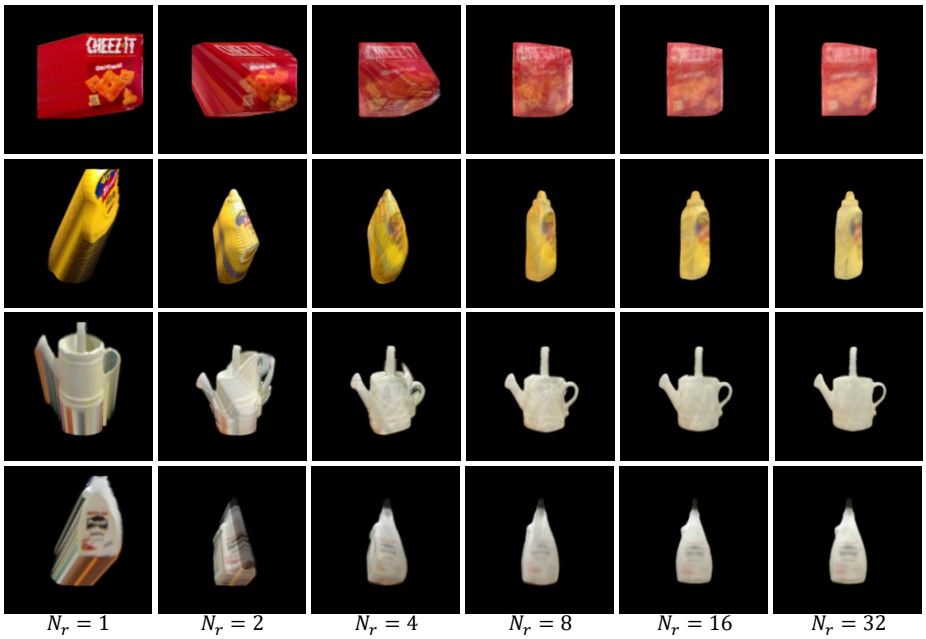

Fig. C.7. Qualitative results of N_r: We visualize the projection of reference feature to compare the quality with the number of reference views. The first and second rows show the reference features from the YCBV dataset object, and the third and last rows visualize the reference feature from the LM dataset object.

References

1. Chen, X., Dong, Z., Song, J., Geiger, A., Hilliges, O.: Category level object pose estimation via neural analysis-by-synthesis. In: European Conference on Computer Vision. pp. 139–156. Springer (2020)
2. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
3. Labbé, Y., Carpentier, J., Aubry, M., Sivic, J.: Cosypose: Consistent multi-view multi-object 6d pose estimation. In: European Conference on Computer Vision. pp. 574–591. Springer (2020)
4. Li, Y., Wang, G., Ji, X., Xiang, Y., Fox, D.: Deepim: Deep iterative matching for 6d pose estimation. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 683–698 (2018)
5. Wang, G., Manhardt, F., Tombari, F., Ji, X.: Gdr-net: Geometry-guided direct regression network for monocular 6d object pose estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 16611–16621 (2021)
6. Zhou, Y., Barnes, C., Lu, J., Yang, J., Li, H.: On the continuity of rotation representations in neural networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5745–5753 (2019)