Supplementary: DISP6D: Disentangled Implicit Shape and Pose Learning for Scalable 6D Pose Estimation

A Overview

The supplementary document is divided into several sections, to provide more details of our design and discussion of experiments mentioned in the main text:

Sec. B presents the detailed network structures, including our AdaIN modulation with ablation on the decoder design by switching the roles of shape and pose codes, as well as the 4D HSH formula for rotational position encoding.

Sec. C includes additional details about the training procedure, including the EMA update of $\mathcal{C}^O$ and training data synthesis.

Sec. D illustrates how we conduct translation and scale estimation for novel objects with unknown physical size (Settings I, III).

Sec. E supplements the discussion for *Ours-per* (Setting I), with more qualitative results, comparison with NOCS under the 3D IoU metric, and comparison with RGBD fusion networks.

Sec. F supplements the discussion for *Ours-all* (Setting III), with qualitative results, comparison with *NOCS-all* and PoseContrast, and the visualization of pose codes for a wine bottle with texture resolving its axial symmetric ambiguity.

Sec. G supplements discussion for setting II on T-LESS, with the visualization of 30 T-LESS models, more qualitative cases and our per-object recall rate for "detection+pose estimation" pipeline (Tab. 1b of main text).

Sec. H reports our evaluation regarding the instance-level pose estimation.

B Detailed Network Structures

We provide the detailed network architectures of encoder E, decoder D^{rgb} and D^{depth}, and conditioned block B in Fig. 1. Furthermore, we explain details of the AdaIN modulation and 4D HSH formula for rotational position encoding in the subsections below.

B.1 AdaIN Modulation (Sec. 3.1 of Main Text)

We use the AdaIN modulation[26] to condition the per-view reconstruction on object code. Specifically, we transform the shape code by $(\boldsymbol{g}_i^s, \boldsymbol{g}_i^b) = FC_i(\boldsymbol{z}_o) \in \mathbb{R}^{2C_i}$ and modulate the intermediate feature map $\mathbf{F}_i \in \mathbb{R}^{C_i \times H_i \times W_i}$ decoded from the pose code by

$$\widetilde{\mathbf{F}}_i = \boldsymbol{g}_i^s \odot \frac{\mathbf{F}_i - \mu(\mathbf{F}_i)}{\sigma(\mathbf{F}_i)} + \boldsymbol{g}_i^b,$$

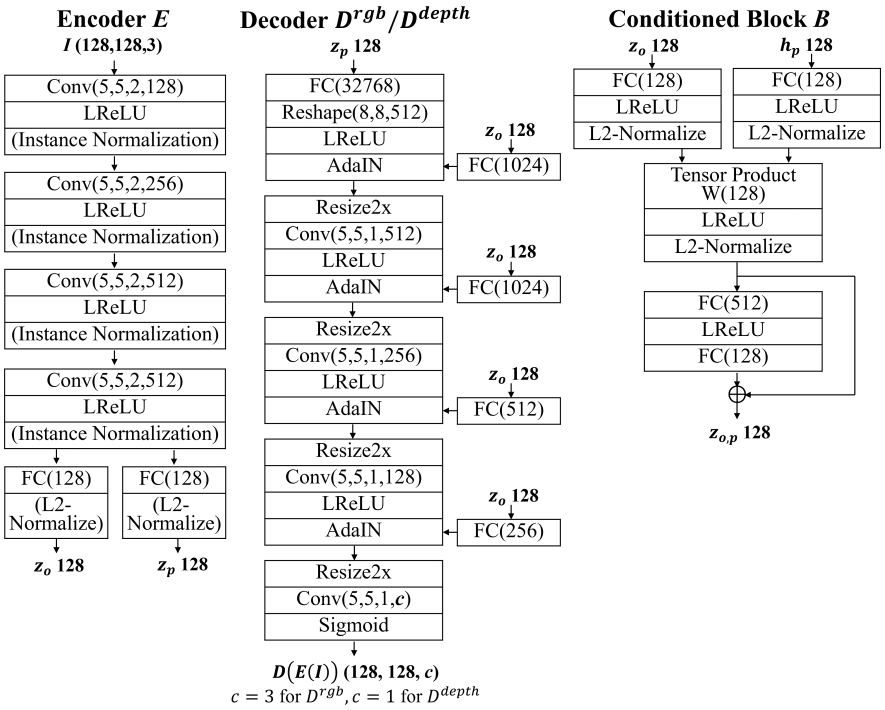


Fig. 1: The detailed network structures. The layer parameters are in the formats: Conv(*filter height, filter width, stride, filter number*), FC(*output dimension*) and W(*output dimension*). For encoder, we use instance normalization (IN) and L2 normalization only when training on the NOCS CAMERA dataset.

where FC_i is a fully connected layer, $\odot$ is the element-wise product, and $\mu(\cdot)$ and $\sigma(\cdot)$ compute the mean and standard deviation vectors across the spatial dimensions, respectively.

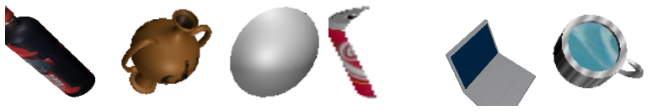
We note that an alternative is to swap the roles of shape $\boldsymbol{z}_o$ and pose $\boldsymbol{z}_p$, and decode from $\boldsymbol{z}_o$ with $\boldsymbol{z}_p$ as the condition in AdaIN modulation. We compare these two configurations by following setting III (Sec. 5.3 of main text) to train on the CAMERA by combining all 6 categories into one set, and report the average precision on REAL275 in Tab. 1.

Tab. 1 verifies our design of using the $\boldsymbol{z}_o$-conditioned decoder: the swapping of $\boldsymbol{z}_o$ and $\boldsymbol{z}_p$ for decoder notably degrades performance across different thresholds.

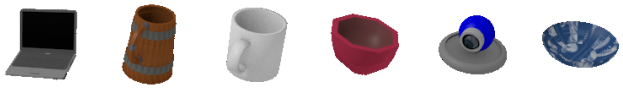
Qualitatively we observe that during training with the $\boldsymbol{z}_p$-modulated decoder, the shape contrastive loss is hard to minimize and as a result, objects cannot find their corresponding representations as nearest neighbors in the latent shape space (see Fig. 2 for examples of training images not retrieving proper objects based on cosine distance Eq. 2 of main text), which also lead to poor scaling to novel objects at the test stage. This performance difference from the $\boldsymbol{z}_o$-modulated decoder can be attributed to the fact that AdaIN modulation changes

Table 1: Ablation study on the decoder design. Reported are mAP at different thresholds of rotation error (in degrees) for mixed categories of REAL275 (setting III)

	AP_5	AP_{10}	AP_{15}	AP_{20}	AP_{30}	AP_{60}
$\boldsymbol{z}_p$-conditioned decoder	2.6	14.7	29.8	43.5	59.9	79.1
$\boldsymbol{z}_o$-conditioned decoder	**9.1**	**30.9**	**50.7**	**64.4**	**75.3**	**84.3**



Query image crops from the CAMERA training set



Retrieved 1NN object, w.r.t using a $\boldsymbol{z}_p$-conditioned decoder

Retrieved 1NN object, w.r.t using a $\boldsymbol{z}_o$-conditioned decoder

Fig. 2: Qualitative results on shape retrieval for image crops from the CAMERA training set, with regard to different decoder designs. For each column, the top row shows the training image, the middle row the retrieved 1-nearest neighbor object by using the $\boldsymbol{z}_p$-conditioned decoder, and the bottom row the nearest neighbor object by using the $\boldsymbol{z}_o$-conditioned decoder.

the overall structure of an image by the spatially uniform affine transformation, which better matches the semantics of shape representation that controls drastic shape variations, rather than the semantics of pose representation that controls the gradual and local variation of viewpoints. By using $\boldsymbol{z}_p$ for overall structural AdaIN modulation, the $\boldsymbol{z}_p$-conditioned decoder effectively forces the shape code to encode both different objects and their subtle view changes simultaneously, which are conflicting aims and lead to many difficulties for learning both shape space and its conditioned pose space.

B.2 4D HSH Formula (Sec. 3.3 of Main Text)

For a rotation $p \in SO(3)$ with in-plane rotation $\beta \in [0, 2\pi]$, zenith $\theta \in [0, \pi]$ and azimuth $\phi \in [0, 2\pi]$, and l, m, n being the polynomial degrees, the corresponding

4D HSH function is constructed as

$$Z_{nl}^{m}(\beta,\theta,\phi) = 2^{l+1/2}\sqrt{\frac{(n+1)\Gamma(n-l+1)}{\pi\Gamma(n+l+2)}}\Gamma(l+1) \sin^{l}\frac{\beta}{2}C_{n-l}^{l+1}(\cos\frac{\beta}{2})Y_{l}^{m}(\theta,\phi)$$

with C_{n-1}^{l+1} as the Gegenbauer polynomials, and the 3D spherical harmonics $Y_l^m(\theta,\phi) \in \mathbb{C}$ as

$$Y_{l}^{m}(\theta,\phi) = \sqrt{\frac{2l+1}{4\pi}\frac{(l-m)!}{(l+m)!}}e^{im\phi}P_{l}^{m}(\cos\theta),$$

where P_l^m is the associated Legendre function. We then follow Alg. 1 to compute each dimension of the rotational position encoding $\boldsymbol{h}_p$ for pose p. We have shifted β with a small delta of 0.05π, as we observe that having $\beta = 0$ among the sampled rotations would lead to $\sin^l\frac{\beta}{2} = 0$ for $l \neq 0$ and thus the full 128 dimensions of $\boldsymbol{h}_p$ become sparse.

Algorithm 1 Generation of the rotational position encoding $\boldsymbol{h}_p$ for rotation $p(\beta,\theta,\phi)$

```
β := β + 0.05π
x := 0
for each n ∈ [0, ..., 6] do
  for each l ∈ [0, ..., n] do
    for each m ∈ [0, ..., l] do
      while the dimension index x ≤ 128 do
        h_p[x] := Re(Z_nl^m(β, θ, φ))
        h_p[x + 1] := Im(Z_nl^m(β, θ, φ))
        x := x + 2
      end while
    end for
  end for
end for
```

C Training Procedure

C.1 EMA Update of $\mathcal{C}^O$ (Sec. 3.2 of Main Text)

For each $\boldsymbol{c}_i \in \mathcal{C}^O$, we maintain for it two variables: $n_i \in \mathbb{R}^+$ and $\boldsymbol{m}_i \in \mathbb{R}^d, d = 128$, with n_i initialized as 1, and each entry of $\boldsymbol{m}_i$ randomly initialized by the

normal distribution $\mathcal{N}(0,1)$. During each SGD iteration, the variables are updated as follows:

$$n_i \leftarrow d_s n_i + (1-d_s)\sum_o \boldsymbol{w}_i^o$$
$$\boldsymbol{m}_i \leftarrow d_s \boldsymbol{m}_i + (1-d_s)\sum_o \boldsymbol{w}_i^o \boldsymbol{z}_o$$
$$\boldsymbol{c}_i \leftarrow \boldsymbol{m}_i / n_i$$

where o iterates over the training objects in a mini-batch, $\boldsymbol{w}^o \in \{0,1\}^{N_O}$ is the target distribution used for the shape contrastive metric learning, and d_s is the exponential decay rate.

C.2 Training Data and Augmentation Strategy

To prepare the training images for T-LESS objects, we sample 92232 rotations from the combination of 36 in-plane rotations and 2562 equidistant spherical views sampled via [20]. With these sampled rotations, we follow AAE[49] and Multipath-AAE[48] to rotate and center the object with a fixed distance along the camera axis ($700mm$), and render the groundtruth images under fixed lighting with a plain background. Note that our rendering uses simple lighting and rasterization, rather than the physically-based renderer (PBR) in [25,12]. We then augment the corresponding encoder input image following Multipath-AAE[48], with random operations including: 1) changing lighting conditions, 2) applying 2D translation and 2D scaling, 3) adding random background images, and 4) tone mapping the color channels.

To train on the CAMERA dataset, we take the training images of the CAMERA dataset. For each instance we use the image patch masked by its groundtruth 2D mask as the encoder input, and further augment the image by following Multipath-AAE to: 1) apply random 2D scaling and 2) randomly adjust brightness for the *camera* category and color channels for other categories. We separately process *camera* and other categories, because for *camera* the color hue is critical for distinguishing poses (*e.g.,* the front side with lens and the back side with display). To prepare the corresponding reconstruction target, we place the 3D model under the groundtruth rotation and a fixed distance along the camera axis (1 *unit*), and render the target image under fixed lighting with a plain background. Moreover, noticing the biased tendency of aligned training cameras to have their handles on the left side, we augment the camera objects by flipping the z-coordinate of camera meshes and putting them into the training set.

D Translation and Scale Estimation for Novel Objects with Unknown Physical Size (Settings I, III)

For novel test objects whose physical sizes are unknown, the RGB-based translation estimation along the camera axis becomes ill-conditioned due to the scale

ambiguity. Therefore, we refer to depth to remove the scale ambiguity and estimate the translation along the camera axis (*i.e.,* T_z), by bounding box size comparison and mean depth comparison between the query depth and the depth reconstructed from D^{depth}. The full 3D translation $\boldsymbol{T} = (T_x, T_y, T_z)^T$ could then be recovered with the pinhole camera model, based on the estimated T_z and the detected 2D bounding box center (t_x, t_y). We illustrate the detailed process below.

Step 1. Transform the observed query depth map $\mathbf{M}$ to the point cloud $\mathbf{O}$ in the query camera coordinate system with the query camera intrinsic K:

$$\mathbf{O} = \mathcal{W}(K, \mathbf{M})$$

where $\mathcal{W}(\cdot, \cdot)$ is the inverse-projection operation to transform the depth map into the 3D point cloud under the camera space. We note that the z-axis is the camera axis and the z-coordinates of the 3D point cloud are equal to the depth map values, while the x, y-coordinates are recovered accordingly by referring to the pinhole camera model with the given camera intrinsic.

We further filter the query $\mathbf{O}$ with outlier removal, where we delete points in $\mathbf{O}$ if they are distant from their k-th (*e.g.,* $k = 100$) nearest neighbors searched in $\mathbf{O}$, with the distance threshold as $5cm$.

Step 2. Transform the reconstructed depth $\mathbf{M}_r = D^{depth}(E(\mathbf{I}))$ to a point cloud $\mathbf{O}_r$ in the object coordinate system, where $\mathbf{I}$ is the query image and $\mathbf{O}_r$ is recovered to be a representation of the visible part for the observed object placed with the observed rotation in its object coordinate system.

To derive $\mathbf{O}_r$ from $\mathbf{M}_r$, we recall that $\mathbf{M}_r$ is supervised to reconstruct a canonical depth map of the object under the observed orientation, whose groundtruth signal is rendered by placing the rotated object at a fixed distance $t_{r,z}$ along the camera axis (*i.e.,* z-axis). Therefore with the training camera intrinsic K_r and $T_{r,z}$, $\mathbf{O}_r$ could be recovered by an inverse process of rendering:

$$\mathbf{O}_r = \mathcal{W}(K_r, \mathbf{M}_r) - (0, 0, T_{r,z})^T$$

Step 3. Estimate the relative scaling factor s from $\mathbf{O}_r$ to $\mathbf{O}$. We compute s as the diagonal ratio of the bounding box along the x, y-axes between $\mathbf{O}$ and $\mathbf{O}_r$. Note that for scale estimation we do not refer to the z-axis, in order to reduce the influence from the noise of depth values.

Step 4. Estimate the translation along the z-axis (*i.e.,* camera axis) T_z with the mean depth comparison

$$T_z = \text{avg}(\mathbf{O}_z) - s\,\text{avg}(\mathbf{O}_{r,z})$$

where $\text{avg}(\mathbf{O}_z), \text{avg}(\mathbf{O}_{r,z})$ respectively denote the average of z-value for $\mathbf{O}$ and $\mathbf{O}_r$.

Step 5. Recover the full translation. We assume the 2D projected bounding box of decoder output is centered, and derive the translation T_x, T_y along the xy-plane with the pinhole camera model as

$$T_x = T_z(t_x - p_x)/f_x$$
$$T_y = T_z(t_y - p_y)/f_y$$

where (t_x, t_y) is the 2D bounding box center obtained from detection, and $(f_x, f_y), (p_x, p_x)$ are the focal and principal point of the query camera intrinsic K.

Step 6. Outlier removal. We note that the mean depth and bounding box size are sensitive to outliers. Therefore we align $s\mathbf{O}_r + \boldsymbol{T}$ with the observed $\mathbf{O}$ and conduct a simple outlier removal for both $\mathbf{O}_r$ and $\mathbf{O}$, where from $\mathbf{O}_r$ we remove point $\boldsymbol{p}_r \in \mathbf{O}_r$ if $s\boldsymbol{p}_r + \boldsymbol{T}$ is distant from the observed depth $\mathbf{O}$, and from $\mathbf{O}$ we remove points that are distant from $s\mathbf{O}_r + \boldsymbol{T}$.

Step 7. Update $s, \boldsymbol{T}$ with the filtered $\boldsymbol{O}_r$ and $\mathbf{O}$. We once again estimate the translation $\boldsymbol{T}$ and scale s by following the procedure described from Step 3 to Step 5 and comparing the filtered $\mathbf{O}_r$ and $\mathbf{O}$.

E Supplementary for *Ours-per* (Sec. 5.2 of Main Text, Setting I)

Qualitative Results We provide in Fig. 7 more qualitative cases of our pose estimation result for *Ours-per* on all 6 scenes of the REAL275 testing set.

3D IoU To supplement Sec. 5.2 of our main text, we report in Fig. 5c the 3D IoU for *Ours-per* and compare it with NOCS[56] (denoted as *NOCS-per* in Fig. 5c). 3D IoU not only evaluates the 6D pose estimation but also takes scale estimation into consideration. We note that our closest prior work [11] neither reports 3D IoU nor discusses on scale estimation.

With a simple mean depth comparison for translation and bounding box size comparison for scale estimation, we observe a comparable average performance between *Ours-per* and NOCS [56] with the 3D IoU metric, while our contribution to rotation estimation is clearly supported by the numerical results on rotation error (Fig. 5a), as is also discussed in the main text.

Comparison with RGBD Fusion Networks To provide a more complete view of different category-level pose estimation methods, we include for comparison more methods that fuse RGBD input by networks [53,6,7,9,35], particularly the state-of-the-art CASS[6], SPD [53], SGPA[7] and DualPoseNet [35] which provide open-source codes. Note that all these methods train pose estimation networks with real data from REAL275[56], which further helps bridge the domain gap between training and testing data. The settings and scopes of different methods are listed in Tab. 2. Fig. 6 shows the mAP of rotation error, translation error and 3D IoU at different thresholds. As shown here, we note that fusing RGB and depth map with a powerful 3D point cloud processing module can significantly boost pose estimation performance, which can be an important augmentation to our simple but scalable pose estimation framework.

F Supplementary for *Ours-all*(Sec. 5.3 of main text, Setting III)

Qualitative Results We provide in Fig. 7 qualitative cases of our pose estimation result for *Ours-all* on all 6 scenes of the REAL275 testing set.

Table 2: Comparing the scopes of different methods on REAL275. All methods follow setting I to assign each category a specific network branch, and use query depth for translation estimation.

	CASS[6]	SPD[53]	DualPoseNet[35]	SGPA[7]	NOCS[56]	Chen et al.[11]	Ours-per
RGB-only network input	×	×	×	×	✓	✓	✓
Synthetic training data only	×	×	×	×	×	✓	✓
RGB-only for rotation estimation	×	×	×	×	×	✓	✓

Table 3: Comparing the scopes between ours and NOCS regarding setting III (*i.e., Ours-all, NOCS-all*) and setting I (*i.e., Ours-per, NOCS-per*). Note that *NOCS-per* is the original NOCS [56] that trains respective NOCS map branches for different categories. All methods refer to query depth for translation estimation.

	NOCS-per[56]	Ours-per	NOCS-all	Ours-all
Synthetic training data only	×	✓	×	✓
RGB-only for rotation estimation	×	✓	×	✓
Uniform loss for categories with different object symmetry	×	✓	×	✓
Extention to cross-category	×	×	✓	✓

Comparison between *Ours-all* and *NOCS-all* To supplement the discussion of *Ours-all*, we train *NOCS-all* for NOCS[56] by using a common NOCS map head for all 6 categories, where we follow the training configuration of NOCS[56] to train on both synthetic data and real data, and adopt the same loss function from NOCS[56] by referring to the category label and processing different object symmetry among different categories. We note that with the design of object-conditioned pose code generalization, *Ours-all* exempts the respective loss designs for different categories and adaptively accommodates various object symmetries. We recap the key differences between *Ours-all*, *NOCS-all* in Tab. 3, where we include also the original NOCS[56](denoted as *NOCS-per*) and *Ours-per* for comparison.

We report the mAP for rotation error, translation error and 3D IoU for pose estimation in Fig. 5. By comparing the two cross-category variants, we observe that *Ours-all* significantly outperforms *NOCS-all* on rotation estimation for all 6 categories, and reports better performance than *NOCS-all* regarding the mean 3D IoU. Since compared with *NOCS-per*, *NOCS-all* uses a shared NOCS map prediction head for different categories, we hypothesize that the inter-categorical shape variances pose significant challenges for the shared NOCS map branch to exploit shape similarity and extract shape consistency across categories for reliable NOCS map prediction. In contrast, *Ours-all* shows competitive performance as discussed in Sec. 5.3 of the main text, which is enabled by the shape space metric learning. Indeed, to address the scalability issue our shape-space contrastive metric learns to model the shape similarity based on instance-level

Table 4: Comparison of rotation estimation with PoseContrast[60] on REAL275 with GT 2D mask. We report rotation accuracy under the error threshold of 30°.

	Bottle	Bowl	Camera	Can	Laptop	Mug	Ave.
PoseContrast[60]	83.5	81.9	11.7	87.2	30.0	35.8	55.0
Ours-all	**96.8**	**99.8**	**58.6**	**99.3**	**93.9**	**91.0**	**89.9**

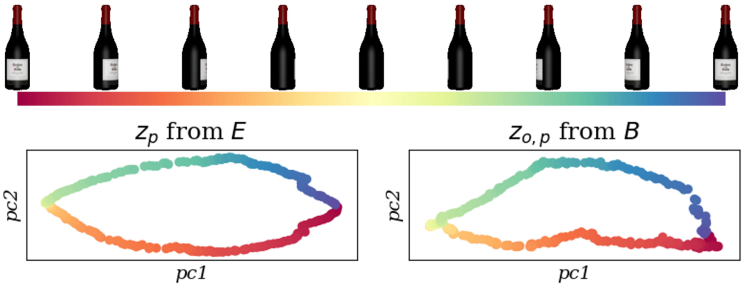


Fig. 3: Visualization of pose codes for a textured wine bottle viewed in 360° as it rotates axially (upper). We show the top two PCA projection of pose codes $\boldsymbol{z}_p, \boldsymbol{z}_{o,p}$ produced by *Ours-all* network (bottom), with point color encoding the viewpoints above. Texture has enabled distinguishing axial symmetries.

shape discrimination, which allows adaptive exploitation for both inter- and intra-categorical shape features without referring to categorical labels.

Comparison between *Ours-all* and PoseContrast We compare with a state-of-the-art PoseContrast[60] that works on the cross-category setting for resolving only the 3D rotation estimation, where we retrain PoseContrast[60] on our setting III of cross-category objects, and evaluate only the rotation accuracy on REAL275 images with GT 2D mask. Results in Table 4 demonstrate our superiority in rotation learning for accommodating categories with different object symmetry, where PoseContrast[60] does not handle objects with different symmetries as well as we do.

Visualization of Pose Codes for the Textured Bottle For symmetric objects with textural features solving the pose ambiguity, we note that our network can indeed tell the different poses by referring to the textures. Taking a textured wine bottle from the CAMERA training objects as an example, we rotate it around its symmetry axis and respectively inspect the top 2 PCA projections for pose codes $\boldsymbol{z}_p, \boldsymbol{z}_{o,p}$. The visualization result is in Fig. 3, where the pose codes well describe the different textural appearance.

Table 5: Object recall rate with $e_{VSD} < 0.3$ for our full 2D detection+pose estimation pipeline, where we train on Obj. 1-18 and report on all 30 objects. Instances with visible portion > 10% for all T-LESS Primesense images are considered.

Obj-id	1	2	3	4	5	6	7	8	9	10
Recall rate	26.05	16.85	33.52	25.43	51.73	47.90	19.53	21.85	32.88	44.50
Obj-id	11	12	13	14	15	16	17	18	19	20
Recall rate	21.14	42.97	41.44	35.28	42.77	41.23	49.06	65.90	22.77	24.09
Obj-id	21	22	23	24	25	26	27	28	29	30
Recall rate	33.22	20.44	18.18	42.32	34.28	45.73	27.42	43.67	52.91	35.66

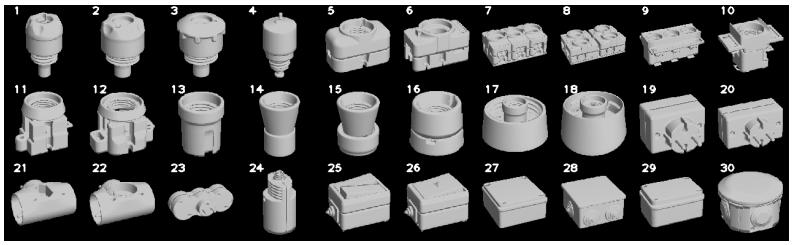


Fig. 4: 30 objects included in the T-LESS dataset [23].

G Results on T-LESS (Sec. 5.4 of main text, Setting II)

30 Objects Included in T-LESS In Fig. 4, we visualize the CAD models of all 30 objects in T-LESS[23]. As we train on Obj. 1-18 while leaving Obj. 19-30 as novel objects in the testing stage, the drastic shape variance and different rotational symmetry can be observed when comparing among the training objects, and comparing between the training objects and the unseen test objects. Our method accommodates these different shapes by a single network.

Per-Object Recall Rate for the Full "Detection+Pose Estimation" Pipeline. For the pose estimation results with 2D bounding boxes detected by MaskR-CNN [17] (Tab. 1b in the main text), we report the detailed recall rate with $e_{VSD} < 0.3$ for each of the 30 objects in Tab. 5. Here we have followed the single object single instance protocol as described in [24].

Qualitative Results for the Full "Detection+Pose Estimation" Pipeline. We provide in Fig. 8 more qualitative cases of our full "detection+pose estimation" pipeline. We notice false negatives caused by failures of 2D detection. However, for the detected instances, our pose estimation could well process both trained and novel objects, even under the challenging conditions of cluttering and partial occlusion.

Table 6: Comparison of pose estimation on T-LESS dataset, following the ViVo setting of BOP Challenge 2020[25]. All reported methods test with single RGB image. † indicates having post-refinement. Bold statistics in black, red, and blue respectively indicate the best, second best, and third best.

Method	Training Image Type	AR_{VSD}	AR_{MSSD}	AR_{MSPD}	AR	Time(s)
AAE[50]	Real+Syn	0.196	0.211	0.504	0.304	**0.194**
Pix2Pose[42]	Real	0.261	0.296	0.476	0.344	1.084
CDPN[34]	Syn	0.303	**0.338**	0.579	0.407	1.849
EPOS[22]	Syn	**0.380**	**0.403**	**0.619**	**0.467**	1.992
CosyPose[32]†	Syn	**0.571**	**0.589**	**0.761**	**0.640**	**0.493**
Ours	Syn	**0.316**	0.326	**0.650**	**0.431**	**0.118**

H Instance-Level Pose Estimation

To explore the limiting case of instance level pose estimation where all objects are used for training, we compare with the state-of-the-art instance-level methods [49,50,42,34,32,22] on the T-LESS dataset, where we follow the setting of BOP Challenge 2020[25] to evaluate our method on the ViVo task (varying number of instances of a varying number of objects) for 6D localization.

Metrics Three pose-error metrics are measured in the BOP challenge[25]: Visible Surface Discrepancy (VSD), Maximum Symmetry-Aware Surface Distance (MSSD), and Maximum Symmetry-Aware Projection Distance (MSPD). These metrics are invariant under symmetry ambiguity.

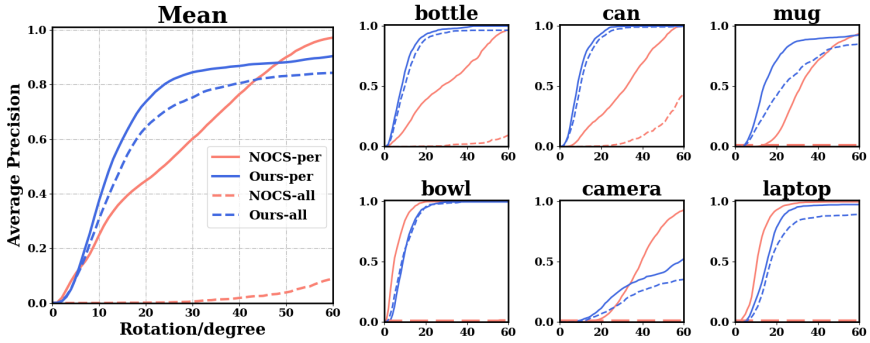
For each metric, we follow the BOP Challenge[25] to calculate the average recall rate under a list of thresholds of correctness (denoted AR_{VSD}, AR_{MSSD}, AR_{MSPD}), as well as the overall average recall $AR = (AR_{VSD} + AR_{MSSD} + AR_{MSPD})/3$.

Training Strategy We train our network on all 30 T-LESS models. Our training set combines synthetic images with 92232 poses per object rendered by the pipeline described in Sec.C.2, and the photorealistic training images from the BOP challenge, which are generated by a physically-based renderer (PBR)[12]. This combination helps us to learn regular latent spaces and to better bridge the synthetic-to-real domain gap. Our 2D detector is the MaskR-CNN adopted from CosyPose [32].

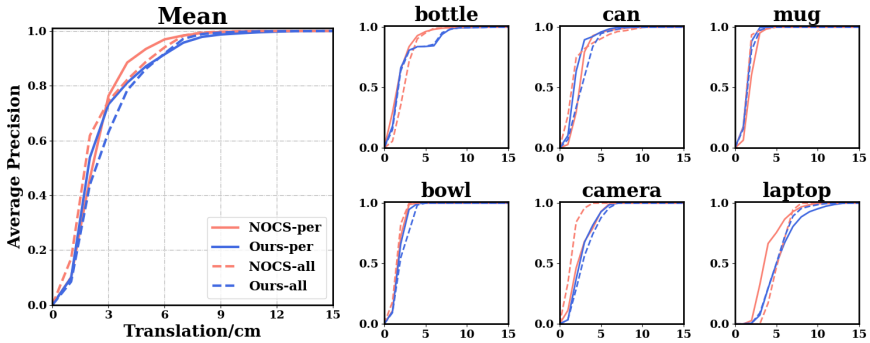
Results and Comparison We test our method on a machine with i7-6700K 4GHz CPU and Nvidia GTX 1080 GPU, and report the performance and the average running time per image in Tab. 6, where we compare with other single RGB-based methods from the BOP leaderboard. Note that CDPN[34], EPOS[22] and CosyPose[32] are trained with synthetic PBR images, while Pix2Pose[42] is trained with real images.

Among the methods listed in Tab. 6, ours is capable of providing a fast yet reliable pose estimation, which could serve as initialization and be further refined if applicable. Specifically, we rank third by the overall AR, and second by AR_{MSPD} which evaluates the 2D projections and thus exempts from the influence of inac-

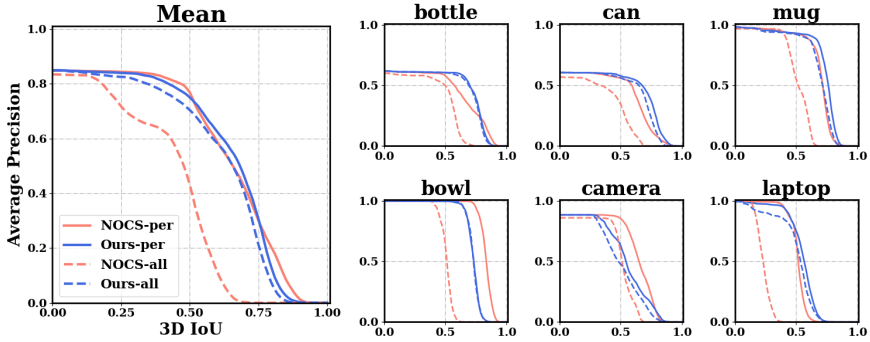
curate depth estimation. Meanwhile, building on an autoencoding pipeline (same as AAE [50]) enables our approach to do fast pose estimation for detected instances, with lower time cost compared with the other methods [42,34,22,32]. Cosypose [32] has the best performance among all error metrics; however, it relies on post-refinement with a regression network after the initial pose estimation. EPOS [22] also outperforms ours overall, as it uses PnP-RANSAC on many-to-many 2D-3D correspondences for reliable rotation and translation with a significant time cost. In comparison, our estimated depth from the pinhole camera model is sensitive to the inaccurate size of the detected 2D bounding box, and we show better performance on AR_{MSPD} where the influence of depth error is minimized.



(a) **Rotation**: AP at different rotation error thresholds

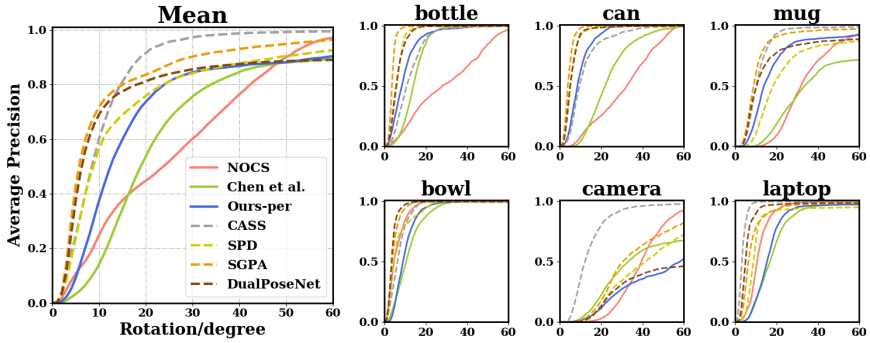


(b) **Translation**:AP at different translation error thresholds

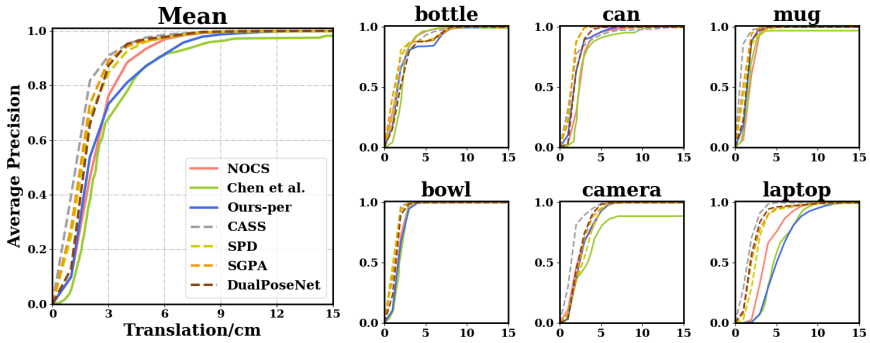


(c) **3D IoU**:AP at different 3D IoU thresholds

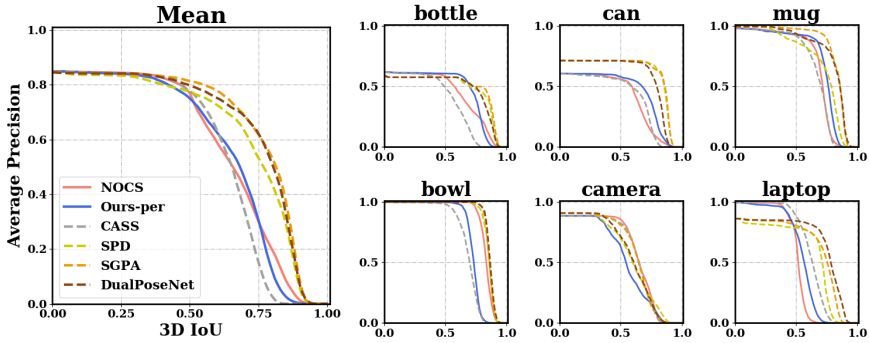
Fig. 5: Comparison on REAL275 for ours and NOCS [56] regarding setting I (*i.e., Ours-per, NOCS-per*) and setting III (*i.e., Ours-all, NOCS-all*). Note that *NOCS-per* is the original NOCS [56] that trains respective NOCS map branches for different categories. Scope of compared methods is listed in Tab. 3. Reported are the average precision under different rotation or translation errors and 3D IoU thresholds.



(a) **Rotation**: AP at different rotation error thresholds



(b) **Translation**:AP at different translation error thresholds



(c) **3D IoU**:AP at different 3D IoU thresholds

Fig. 6: Comparison on REAL275 regarding setting I. Scope of compared methods is listed in Tab. 2. Reported are the average precision under different rotation or translation error and 3D IoU thresholds. CASS [6], SPD [53], SGPA [7] and DualPoseNet [35] fuse RGBD input by networks and train with real data.

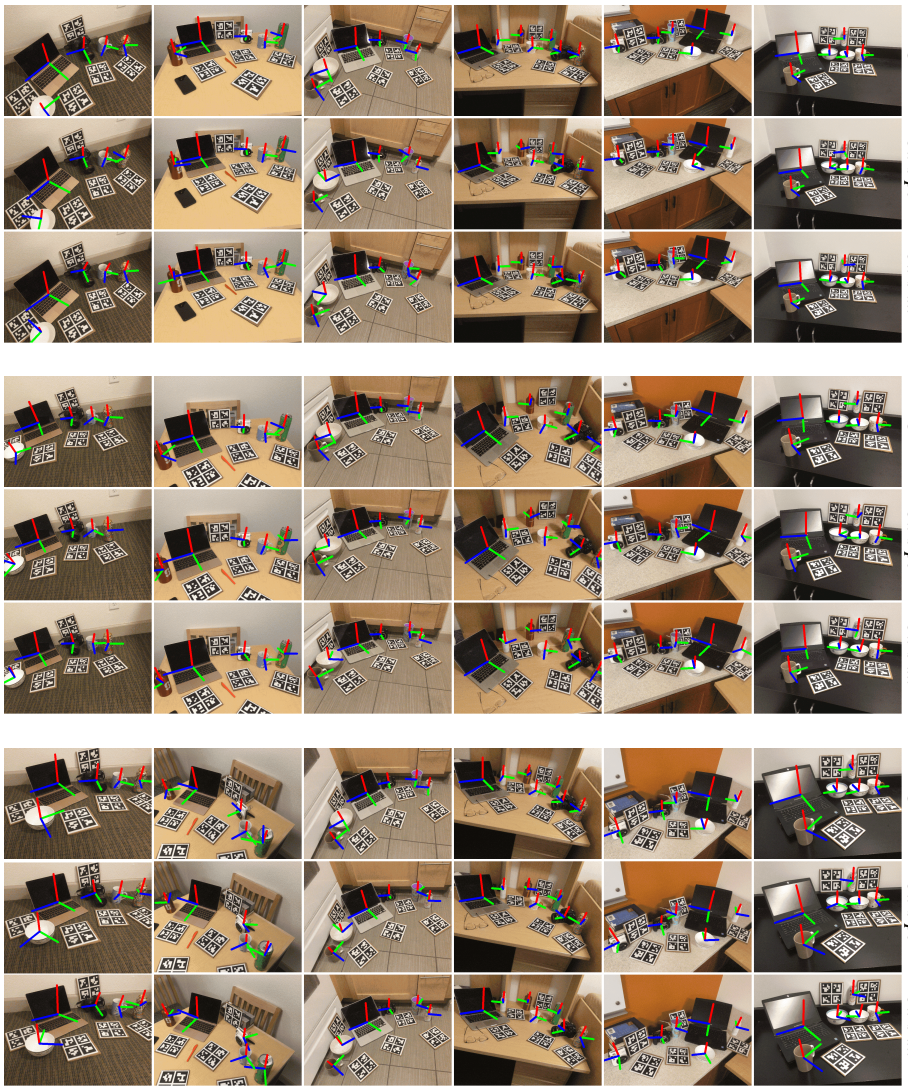


Fig. 7: Qualitative results on the 6 test scenes of REAL275 for *Ours-per*(Setting I) and *Ours-all*(Setting III). Images of a column belong to a scene.

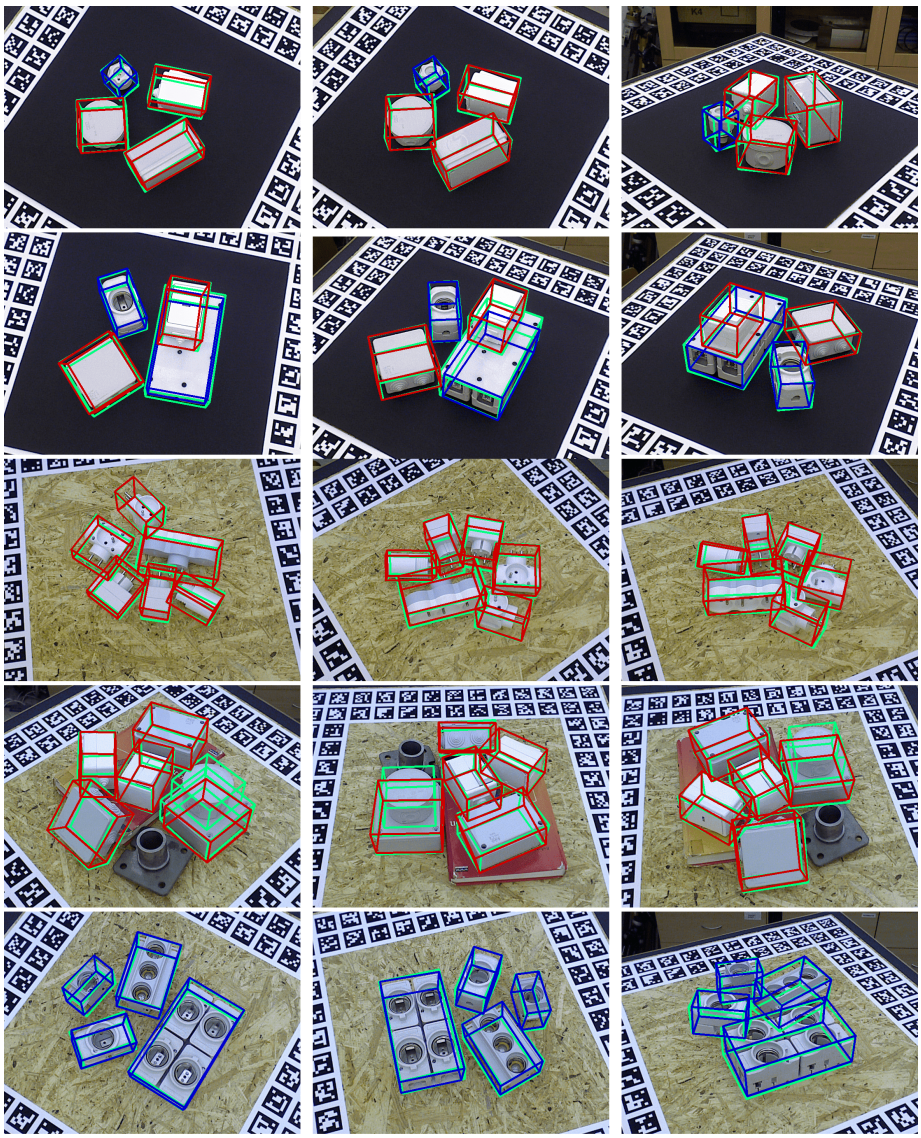


Fig. 8: Qualitative cases on T-LESS of setting II. Blue and red boxes respectively indicate our estimation on trained objects and unseen objects, with green box indicating the groundtruth.

References

1. Bengio, Y., Courville, A., Vincent, P.: Representation learning: A review and new perspectives. IEEE Transactions on Pattern Analysis and Machine Intelligence **35**(8) (2013)
2. Billings, G., Johnson-Roberson, M.: Silhonet: An rgb method for 3d object pose estimation and grasp planning. arXiv preprint arXiv:1809.06893 (2018)
3. Bouchacourt, D., Ibrahim, M., Deny, S.: Addressing the topological defects of disentanglement via distributed operators (2021)
4. Brachmann, E., Michel, F., Krull, A., Ying Yang, M., Gumhold, S., et al.: Uncertainty-driven 6d pose estimation of objects and scenes from a single rgb image. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3364–3372 (2016)
5. Chang, A.X., Funkhouser, T., Guibas, L., Hanrahan, P., Huang, Q., Li, Z., Savarese, S., Savva, M., Song, S., Su, H., Xiao, J., Yi, L., Yu, F.: Shapenet: An information-rich 3d model repository (2015)
6. Chen, D., Li, J., Wang, Z., Xu, K.: Learning canonical shape space for category-level 6d object pose and size estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (June 2020)
7. Chen, K., Dou, Q.: Sgpa: Structure-guided prior adaptation for category-level 6d object pose estimation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 2773–2782 (2021)
8. Chen, T., Kornblith, S., Norouzi, M., Hinton, G.: A simple framework for contrastive learning of visual representations. In: International conference on machine learning. pp. 1597–1607. PMLR (2020)
9. Chen, W., Jia, X., Chang, H.J., Duan, J., Shen, L., Leonardis, A.: Fs-net: Fast shape-based network for category-level 6d object pose estimation with decoupled rotation mechanism. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1581–1590 (2021)
10. Chen, X., Duan, Y., Houthooft, R., Schulman, J., Sutskever, I., Abbeel, P.: Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In: Advances in Neural Information Processing Systems 29, pp. 2172–2180 (2016)
11. 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 (ECCV) (2020)
12. Denninger, M., Sundermeyer, M., Winkelbauer, D., Zidan, Y., Olefir, D., Elbadrawy, M., Lodhi, A., Katam, H.: Blenderproc. arXiv preprint arXiv:1911.01911 (2019)
13. Dumoulin, V., Perez, E., Schucher, N., Strub, F., Vries, H.d., Courville, A., Bengio, Y.: Feature-wise transformations. Distill **3**(7), e11 (2018)
14. González, Á.: Measurement of areas on a sphere using fibonacci and latitude–longitude lattices. Mathematical Geosciences **42**(1), 49 (2010)
15. Grabner, A., Roth, P.M., Lepetit, V.: Location field descriptors: Single image 3d model retrieval in the wild. In: 2019 International Conference on 3D Vision (3DV). pp. 583–593. IEEE (2019)
16. He, K., Fan, H., Wu, Y., Xie, S., Girshick, R.: Momentum contrast for unsupervised visual representation learning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2020)

17. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask r-cnn. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV). pp. 2961–2969 (2017)
18. Higgins, I., Amos, D., Pfau, D., Racaniere, S., Matthey, L., Rezende, D., Lerchner, A.: Towards a definition of disentangled representations. arXiv preprint arXiv:1812.02230 (2018)
19. Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X., Botvinick, M., Mohamed, S., Lerchner, A.: beta-vae: Learning basic visual concepts with a constrained variational framework. In: ICLR (2017)
20. Hinterstoisser, S., Benhimane, S., Lepetit, V., Fua, P., Navab, N.: Simultaneous recognition and homography extraction of local patches with a simple linear classifier. In: BMVC. pp. 1–10 (2008)
21. Hinterstoisser, S., Lepetit, V., Ilic, S., Holzer, S., Bradski, G., Konolige, K., Navab, N.: Model based training, detection and pose estimation of texture-less 3d objects in heavily cluttered scenes. In: Asian conference on computer vision. pp. 548–562. Springer (2012)
22. Hodaň, T., Baráth, D., Matas, J.: EPOS: Estimating 6D pose of objects with symmetries. IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2020)
23. Hodaň, T., Haluza, P., Obdržálek, Š., Matas, J., Lourakis, M., Zabulis, X.: T-LESS: An RGB-D dataset for 6D pose estimation of texture-less objects. IEEE Winter Conference on Applications of Computer Vision (WACV) (2017)
24. Hodan, T., Michel, F., Brachmann, E., Kehl, W., GlentBuch, A., Kraft, D., Drost, B., Vidal, J., Ihrke, S., Zabulis, X., et al.: Bop: Benchmark for 6d object pose estimation. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 19–34 (2018)
25. Hodaň, T., Sundermeyer, M., Drost, B., Labbé, Y., Brachmann, E., Michel, F., Rother, C., Matas, J.: BOP challenge 2020 on 6D object localization. European Conference on Computer Vision Workshops (ECCVW) (2020)
26. Huang, X., Belongie, S.: Arbitrary style transfer in real-time with adaptive instance normalization. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 1501–1510 (2017)
27. Karras, T., Laine, S., Aila, T.: A style-based generator architecture for generative adversarial networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4401–4410 (2019)
28. Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J., Aila, T.: Analyzing and improving the image quality of stylegan. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8110–8119 (2020)
29. Kehl, W., Manhardt, F., Tombari, F., Ilic, S., Navab, N.: Ssd-6d: Making rgb-based 3d detection and 6d pose estimation great again. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 1521–1529 (2017)
30. Khemakhem, I., Kingma, D., Monti, R., Hyvarinen, A.: Variational autoencoders and nonlinear ica: A unifying framework. In: Proceedings of International Conference on Artificial Intelligence and Statistics. Proceedings of Machine Learning Research, vol. 108. PMLR (2020)
31. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
32. 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)
33. Lepetit, V.: Recent advances in 3d object and hand pose estimation. arXiv preprint arXiv:2006.05927 (2020)

34. Li, Z., Wang, G., Ji, X.: Cdpn: Coordinates-based disentangled pose network for real-time rgb-based 6-dof object pose estimation. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 7678–7687 (2019)
35. Lin, J., Wei, Z., Li, Z., Xu, S., Jia, K., Li, Y.: Dualposenet: Category-level 6d object pose and size estimation using dual pose network with refined learning of pose consistency. arXiv preprint arXiv:2103.06526 (2021)
36. Locatello, F., Abbati, G., Rainforth, T., Bauer, S., Schölkopf, B., Bachem, O.: On the fairness of disentangled representations. In: Advances in Neural Information Processing Systems. vol. 32. Curran Associates, Inc. (2019)
37. Locatello, F., Bauer, S., Lucic, M., Raetsch, G., Gelly, S., Schölkopf, B., Bachem, O.: Challenging common assumptions in the unsupervised learning of disentangled representations. In: Chaudhuri, K., Salakhutdinov, R. (eds.) Proceedings of International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 97. PMLR (2019)
38. Martyn, J., Vidal, G., Roberts, C., Leichenauer, S.: Entanglement and tensor networks for supervised image classification (2020)
39. Nguyen, V.N., Hu, Y., Xiao, Y., Salzmann, M., Lepetit, V.: Templates for 3d object pose estimation revisited: Generalization to new objects and robustness to occlusions. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 6771–6780 (June 2022)
40. van den Oord, A., Li, Y., Vinyals, O.: Representation learning with contrastive predictive coding. arXiv preprint arXiv:1807.03748 (2018)
41. Park, K., Mousavian, A., Xiang, Y., Fox, D.: Latentfusion: End-to-end differentiable reconstruction and rendering for unseen object pose estimation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2020)
42. Park, K., Patten, T., Vincze, M.: Pix2pose: Pixel-wise coordinate regression of objects for 6d pose estimation. arXiv preprint arXiv:1908.07433 (2019)
43. Pasha Hosseinbor, A., Chung, M.K., Koay, C.G., Schaefer, S.M., van Reekum, C.M., Schmitz, L.P., Sutterer, M., Alexander, A.L., Davidson, R.J.: 4d hyperspherical harmonic (hyperspharm) representation of surface anatomy: A holistic treatment of multiple disconnected anatomical structures. Medical Image Analysis **22**(1), 89–101 (2015)
44. Peng, S., Liu, Y., Huang, Q., Zhou, X., Bao, H.: Pvnet: Pixel-wise voting network for 6dof pose estimation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4561–4570 (2019)
45. Pitteri, G., Bugeau, A., Ilic, S., Lepetit, V.: 3D Object Detection and Pose Estimation of Unseen Objects in Color Images with Local Surface Embeddings. In: 15th Asian Conference on Computer Vision. Kyoto (virtual conference), Japan (Nov 2020)
46. Platanios, E.A., Sachan, M., Neubig, G., Mitchell, T.: Contextual parameter generation for universal neural machine translation. In: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. pp. 425–435 (2018)
47. Rad, M., Lepetit, V.: Bb8: A scalable, accurate, robust to partial occlusion method for predicting the 3d poses of challenging objects without using depth. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 3828–3836 (2017)
48. Sundermeyer, M., Durner, M., Puang, E.Y., Marton, Z.C., Vaskevicius, N., Arras, K.O., Triebel, R.: Multi-path learning for object pose estimation across domains. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 13916–13925 (2020)

49. Sundermeyer, M., Marton, Z.C., Durner, M., Brucker, M., Triebel, R.: Implicit 3d orientation learning for 6d object detection from rgb images. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 699–715 (2018)
50. Sundermeyer, M., Marton, Z.C., Durner, M., Triebel, R.: Augmented autoencoders: Implicit 3d orientation learning for 6d object detection. International Journal of Computer Vision pp. 1–16 (2019)
51. Tekin, B., Sinha, S.N., Fua, P.: Real-time seamless single shot 6d object pose prediction. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 292–301 (2018)
52. Tenenbaum, J.B., Freeman, W.T.: Separating style and content with bilinear models. Neural Computation **12**(6), 1247–1283 (2000). https://doi.org/10.1162/089976600300015349
53. Tian, M., Ang Jr, M.H., Lee, G.H.: Shape prior deformation for categorical 6d object pose and size estimation. In: Proceedings of the European Conference on Computer Vision (ECCV) (August 2020)
54. Tian, Z., Shen, C., Chen, H.: Conditional convolutions for instance segmentation. In: Proc. Eur. Conf. Computer Vision (ECCV) (2020)
55. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L.u., Polosukhin, I.: Attention is all you need. In: Advances in Neural Information Processing Systems. vol. 30. Curran Associates, Inc. (2017)
56. Wang, H., Sridhar, S., Huang, J., Valentin, J., Song, S., Guibas, L.J.: Normalized object coordinate space for category-level 6d object pose and size estimation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2642–2651 (2019)
57. Wen, Y., Pan, H., Yang, L., Wang, W.: Edge enhanced implicit orientation learning with geometric prior for 6d pose estimation. IEEE Robotics and Automation Letters (IROS) **5**(3) (2020)
58. Wu, Z., Xiong, Y., Yu, S.X., Lin, D.: Unsupervised feature learning via non-parametric instance discrimination. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 3733–3742 (2018)
59. Xiang, Y., Schmidt, T., Narayanan, V., Fox, D.: Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes. arXiv preprint arXiv:1711.00199 (2017)
60. Xiao, Y., Du, Y., Marlet, R.: Posecontrast: Class-agnostic object viewpoint estimation in the wild with pose-aware contrastive learning. In: 2021 International Conference on 3D Vision (3DV). pp. 74–84 (2021). https://doi.org/10.1109/3DV53792.2021.00018
61. Zhang, H., Cao, Q.: Detect in rgb, optimize in edge: Accurate 6d pose estimation for texture-less industrial parts. In: 2019 International Conference on Robotics and Automation (ICRA). pp. 3486–3492. IEEE (2019)
62. Zhao, L.: Spherical and spheroidal harmonics: Examples and computations (2017)
63. Zhou, X., Karpur, A., Luo, L., Huang, Q.: Starmap for category-agnostic keypoint and viewpoint estimation. In: European Conference on Computer Vision (ECCV) (2018)