# Zero-Shot Category-Level Object Pose Estimation: Supplementary Material

Walter Goodwin[1⋆], Sagar Vaze[2⋆], Ioannis Havoutis[1], and Ingmar Posner[1]

[1] Oxford Robotics Institute, University of Oxford
[2] Visual Geometry Group, University of Oxford
firstname@robots.ox.ac.uk

## 1 Supplementary Material

In this appendix, we first discuss our choice of dataset, followed by our choice of evaluation categories and sequences, and a description of our pose-labelling procedure, and data pre-processing steps. We then present several further experiments and ablations to our method, showing that performance improves further under greater numbers of target views, and the effectiveness of our full method in refining a pose estimation. Results around the number and diversity of correspondences are given, and the approach to the rigid body transform solution and RANSAC is described further and justified. We examine our choice of evaluation metric for the SO(3) component of pose estimation, and explore the effect of near-symmetries on our results in this light. We give further implementation details on several baselines.

## A CO3D dataset

### A.1 Choice of dataset

A comparison of several multi-category, multi-instance datasets is given in table A.1. Several existing canonical category-level pose datasets are not appropriate for our method as they do not include depth information [46, 47], or only have extremely sparse depth [2]. The **Redwood** dataset [14] contains a good diversity of object categories and instances, with many views per object and ground truth depth maps, but structure-from-motion (SfM) is only run on a small subset of categories and sequences, so very few sequences have camera extrinsics, required to evaluate the multiple target view version of our method. The **REAL275** dataset [45], being motivated in the same embodied settings as the present work, has the appropriate depth and extrinsic information. However, the dataset contains only 6 categories and a small number of instances (7 per category). The present work considers a zero-shot approach to category-level pose, and a strong quantitative and qualitative evaluation of this method requires a large diversity of object categories. **CO3D** [35] provides this, with 51

[⋆] These authors contributed equally

| Dataset | # Cat. | # Obj./Cat. | # View/Obj. | Pose | Extrinsics | Depth | Pcd/Mesh |
|---|---|---|---|---|---|---|---|
| **Pascal3D** [47] | 12 | ∼3000 | 1 | Yes | No | No | No |
| **ObjectNet3D** [46] | 100 | 2019 | 1 | Yes | No | No | No |
| **Objectron** [2] | 9 | 1621 | 268 | Yes | Yes | No* | No* |
| **Redwood** [14] | 320 | ∼28 | ∼2300 | No | No | Yes | No† |
| **REAL275** [45] | 6 | 7 | ∼950‡ | Yes | Yes | Yes | Yes |
| **CO3D** [35] | 51 | ∼380 | ∼79 | No | Yes | Yes | Yes |

Table A.1: A comparison of multi-view category-level datasets (**# Cat.** = number of categories, **# Obj./Cat** = average number of distinct object instances per category, **# View/Obj.** = average number of views of each distinct instance). We find that **CO3D** is the only dataset that offers a large number of categories, with diversity within the category, alongside multiple views and depth information for each object. *Depth and point cloud information in **Objectron** is only available via the highly sparse points used in the SfM process. †The Redwood dataset provides high quality mesh reconstructions for just 398 object instances, from a subset of only 9 categories. ‡The combined train/val/test splits of **REAL275** contain 8,000 frames, each with at least 5 objects present. With 42 object instances, this gives ∼950 appearances per instance.

object categories, each containing a large variety of instances, with depth and camera extrinsic information. While unlike most of the other datasets considered in table A.1, CO3D does not contain labelled category-level pose, we find that we are able to label sufficient sequences ourselves to reach robust quantitative evaluation of our methods and baselines (appendix A.3). As our method is fully unsupervised, we do not require a large labelled dataset for training: a sufficient test set is all that is needed.

### A.2 Choice of evaluation categories & sequences

The CO3D dataset contains hundreds of sequences for each of 51 distinct object categories. In this work, our quantitative evaluation is performed on a subset of 20 of these categories. We ***exclude*** categories based on the following criteria:

- Categories for which the object has one or more axes of infinite rotational symmetry. 16 categories (*apple, ball, baseball bat, bottle, bowl, broccoli, cake, carrot, cup, donut, frisbee, orange, pizza, umbrella, vase, wineglass*).
- Categories for which the object has more than one rotational symmetry. 6 categories (*bench, hot dog, kite, parking meter, skateboard, suitcase*).
- Categories for which an insufficient number of sequences ($< 10$) have high-quality point clouds and camera viewpoints. 6 categories (*banana, cellphone, couch, microwave, stop sign, TV*).
- Categories for which between-instance shape difference made labelling challenging or fundamentally ambiguous. 3 categories (*baseball glove, plant, sandwich*).

This leaves 20 categories, as shown in fig. A.5. Some included categories were still ‘marginal’ under these criteria, for instance *handbag*, where there was a 180º rotational symmetry for most instances. Here, the labelling convention was to, where possible, disambiguate pose labels by which side of the handbag the handle fell onto. Nonetheless, categories such as *handbag* and *toaster* elicited bi-modal predictions from our method, reflecting these ambiguities, as shown in fig. A.5.

We further select a subset of sequences for labelling (appendix A.3) from each of these 20 categories. CO3D provides predicted quality scores for camera viewpoints and point clouds reconstructed by the COLMAP structure-from-motion (SfM) processes [35]. Each category has an average of 356 sequences (distinct object instances), ranging from 21 for *parking meter* to 860 for *backpack*. We choose to consider all sequences that have a viewpoint quality score of more than 1.25, and a point cloud quality of greater than 0.3. On average, this is the top 16% of sequences within a category, and returns a median of 36 valid sequences per category. For our chosen categories (appendix A.2), we choose to label the top 10 sequences based on point cloud scores with category-level pose.

### A.3 Labelling pose for evaluation

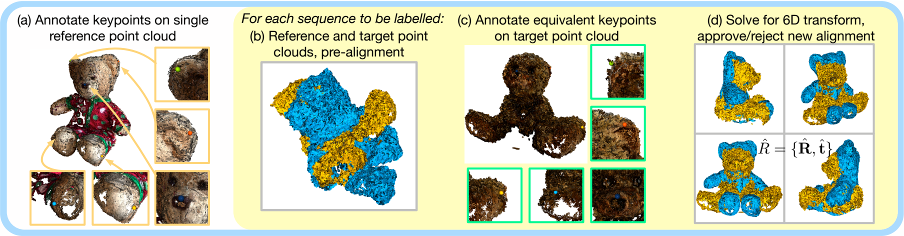
(a) Annotate keypoints on single reference point cloud
For each sequence to be labelled:
(b) Reference and target point clouds, pre-alignment
(c) Annotate equivalent keypoints on target point cloud
(d) Solve for 6D transform, approve/reject new alignment
$\hat{R} = \{\hat{\mathbf{R}}, \hat{\mathbf{t}}\}$

Fig. A.1: The process used in this work to generate category-level pose labels for the CO3D dataset, in the presence of large between-instance shape and appearance shift. Our interface uses Open3D [52] for annotation and visualisation.

The per-frame camera extrinsics in CO3D are given relative to the first frame in each sequence. Thus, the camera extrinsic positions do not relate the SE(3) poses of objects within a category with respect to any category-level canonical pose. Indeed, this is noted by the dataset’s authors [35] as a limitation of using the dataset to learn category-level object representations. To overcome this and enable quantitative evaluation, we design a labelling interface that leverages the sequence point clouds for fast and intuitive category-level pose alignment. The process is depicted in fig. A.1. For each category, we choose the sequence with the highest point cloud quality score to be the reference object. Four or more semantically salient keypoints that are deemed likely to exist, in a spatially consistent manner, across all instances in the category are selected interactively on this point cloud, using the interface. Subsequently, the labeller is presented with the other candidate objects in turn, and selects the equivalent points in the same order. Umeyama's method is then used to solve for the rigid body transform and uniform scaling, given annotated keypoint correspondences [42]. The labeller is then presented with the reference point cloud, overlaid with the transformed target point cloud, both coloured uniformly for clarity, and can inspect the quality of the alignment. If it is adequate, the transform is accepted, and the rigid body parameters $\hat{T} = \left(\hat{\mathbf{R}}, \hat{\mathbf{t}}\right)$ saved as a pose label relative to the target sequence. This provides labels of pose offsets at the point cloud level, which is in world coordinate space. Every frame in a sequence is related to the world coordinates via the predicted camera extrinsics. Further, every sequence will have a relative pose against the reference sequence's point cloud. Using this, a ground-truth relative pose in the camera frame, which is what our method predicts, can be constructed for any two frames $i$ and $j$ from any two sequences $a$ and $b$ as:

$$\mathbf{T}_{a_i b_j} = (\mathbf{T}^{\text{cam}}_{a_i})^{-1} \circ \mathbf{T}^{-1}_{0a} \circ \mathbf{T}_{0b} \circ \mathbf{T}^{\text{cam}}_{b_j} \tag{1}$$

Where $\mathbf{T}$ denotes a $4 \times 4$ homogeneous transform matrix composed from rotation $\mathbf{R}$ and translation $\mathbf{t}$, and $\mathbf{T}_{0a}$, $\mathbf{T}_{0b}$ are the transforms from reference to target object point clouds as computed in our labelling procedure, and $\mathbf{T}^{\text{cam}}_{a_i}$, $\mathbf{T}^{\text{cam}}_{b_j}$ are the camera extrinsics (world to view transforms) from the SfM procedure in CO3D. $\circ$ denotes function composition - as these functions are transformation matrices, the resultant transform is $\mathbf{T}^{\text{cam}}_{b_j}\mathbf{T}_{0b}\mathbf{T}^{-1}_{0a}(\mathbf{T}^{\text{cam}}_{a_i})^{-1}$.

### A.4 Data processing

**Depth completion** CO3D uses crowd-sourced video, with the original data coming from RGB cameras before structure-from-motion is extracted by COLMAP [37]. CO3D chooses to scale all point clouds to have unit standard deviation averaged across 3 world coordinate axes, which then fixes the camera intrinsics and depth maps to be relative to this world coordinate scheme. For our purposes, this scale ambiguity is acceptable - we can nonetheless evaluate SE(3) pose predictions, for which the rotation component is independent of scale, and for which the translation component will be affected but still has a well-posed and recoverable ground truth.

On the other hand, the depth maps in CO3D are estimates from COLMAP's multi-view stereo (MVS) algorithm, and are incomplete. Our method requires accurate depth to project the discovered semantic correspondences into 3D space, enabling a solution for the rigid body transform between object instances (section 4.3). One approach would be to disregard those correspondences that land on an area with unknown depth. However, as the correspondences are found at the ViT patch level ($8 \times 8$ pixels, see section 4.1), we found a small number of missing areas in the per-pixel depth maps led to throwing away a disproportionate amount of correspondences. Instead, we use a fast in-painting method based on the Navier-Stokes equations [5], implemented in OpenCV, to fill missing values.

**Object crops** CO3D uses a supervised segmentation network to produce probabilistic mask labels for every frame. We threshold these and pad the result by 10% to give a region of interest for the objects. We use this to crop the depth maps and RGB images when evaluating our method. However, we do not use these masks further within our method.

# B Further experiments

## B.1 Number of target views

| | Best view only | | Full method | |
|---|---|---|---|---|
| Target views | Acc30 ↑ | Acc15 ↑ | Acc30 | Acc15 |
| 1 | 12.6 | 3.7 | **23.8** | **13.4** |
| 3 | 26.1 | 8.0 | **38.9** | **26.2** |
| 5 | 35.4 | 10.6 | **46.3** | **31.1** |
| 10 | 45.0 | 16.2 | **52.5** | **38.2** |
| 20 | 47.0 | 18.6 | **52.1** | **38.3** |

Table A.2: (Acc30, Acc15 = percentage of predictions with a geodesic error of less than 30º, 15º.). An extension of the comparison in sec. 5.5 of the effect of increasing the number of available target views, and the improvement of the full method including solving for a rigid body transformation, over just taking the best view as a pose prediction.

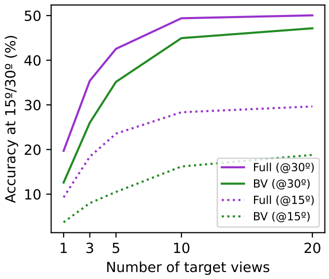

Fig. A.2: 'Full' method (purple) vs 'BV' (best view) only (green). As the number of target views increases, both accuracy metrics improve, though exhibit diminishing returns. The full method leads the best-view ablation throughout, especially in Acc15.

In the main paper, we show that the number of available target views is an important parameter in our method, demonstrating that as we increase from 1 view to 3 and 5 views, pose estimation performance improves. Here, we include two further results, in which 10 and 20 target views are available, to investigate whether these effects continue to scale. We also present the results achieved by taking the coarse pose estimate given by our best view selection method, without further refinement from the rigid body transform component. The results are shown in Tab. A.2 and Fig. A.2. Clearly, increasing the number of target views available has a positive effect on performance, though in an embodied setting this would come at the cost of the time to explore and image multiple views. While it can be seen that by doubling from 5 to 10 target views improves the Acc30 by over 6%, we chose to report only the figures for the small number of views (1, 3, 5) in the main text, to reflect such a practical use case. It can also be seen - as already noted in section 5.5 - that the full method, including the rigid body transform computed leveraging the semantic correspondences, outperforms the baseline of simply taking the 'best' view as predicted by our method's first stage. This continues to hold in the regimes with 10 and 20 target views. Finally, inspecting Fig. A.2 makes it clear that while the full method benefits Acc30, its effect is most marked in improving Acc15 over the performance of taking the best view. This is in line with intuition, which is that the rigid body solution provides fine-tuning on top of a coarse initial estimate (see section 5.5).

### B.2 Design of correspondence method

| Method | Med. Err (↓) | Acc30 (↑) | Acc15 (↑) |
|---|---|---|---|
| Optimal Transport [30] | 54.6 | 44.8 | 29.1 |
| Dual Softmax [13] | 54.0 | 44.5 | 30.1 |
| Cyclical Distances (Ours) | **53.8** | **46.3** | **31.1** |

Table A.3: We ablate our design of correspondence method, which is based on building a cyclical distance map (see Sec. 4.1). Here, we report results of pose estimation using an equivalent map arising from running optimal transport [30] and dual-softmax [13] on top of the raw feature similarity matrix.

Our method builds a cyclical distance map to extract semantic correspondences between two images. Here, we experiment with alternate methods of identifying the corresponding locations. Specifically, our problem is a special case of the general machine learning problem of identifying matches between two sets of features (i.e the spatial features in the target and reference images). As such, given a matrix of feature distances between the two normalised feature maps, $D \in \{-1...1\}^{H' \times W' \times H' \times W'}$, we experiment with two additional methods for selecting $K$ entries to serve as correspondences. In table A.3, we experiment both with a standard optimal transport solution [30] as well as a recent dual-softmax method [13], and find our 'cyclical distance' design choice is the most suitable for this task.

### B.3 Number and diversity of correspondences

In section 4.1, we describe our approach to guarantee the return of a desired number of correspondences through the introduction of the concept of the 'cyclical distance' induced by following a chain of descriptor nearest neighbours from reference image, to target, and back to the reference image. We keep the top-K correspondences under this metric for our method. In some cases, however, there can be a particular region of the two objects that gives rise to a large portion of the top-K correspondences. This can in turn lead to less appropriate pose estimates from the rigid body transform solution (see appendix B.4), as a

| | Without K-means | | With K-means | |
|---|---|---|---|---|
| # Correspondences | Acc30 (↑) | Acc15 (↑) | Acc30 (↑) | Acc15 (↑) |
| 30 | 38.7 | 25.2 | 43.5 | 29.0 |
| 50 | 42.0 | 28.6 | 46.3 | **31.1** |
| 70 | 43.4 | 30.4 | **47.1** | 30.8 |

Fig. A.3: Comparison of results over 20 categories as the number of correspondences is varied, and when K-means clustering is used to return a set of correspondences that are maximally distinct in descriptor space.

transform can produce this cluster of points and give a large number of inliers for RANSAC, while not aligning the object's in a satisfactory global way. To address this bias, we seek to augment the choice of the top-K correspondences to encourage spatial and semantic diversity. Inspired by [4], we employ k-means clustering in descriptor space. We sample the top-$2K$ correspondences under the cyclical distance measure, then seek to produce $K$ clusters. We return a correspondence from each cluster, choosing the one that has the highest ViT salience in the reference image. The effect of this K-means step, and the impact of using differing numbers of correspondences, is shown in fig. A.3. We find that k-means clustering improves performance, and use this throughout the other experiments in this paper. We find that using 50 correspondences in our method is sufficient for a trade-off between run-time, correspondence quality, and pose prediction error.

### B.4 Rigid body transform solution

**Algorithm choice** In our method, given a number of corresponding points in 3D space, we solve for the rigid body transform that minimises the residual errors between the points of the target object, and the transformed points of the reference object. There are a number of solutions to this problem, with some based on quaternions, and some on rotation matrices and the singular value decomposition. A comparison of four approaches is given in [17]. We choose to use Umeyama's method [42], as it allows for simultaneously solving for both the 6D rigid body transform, as well as a uniform scaling parameter. It is also robust under large amounts of noise, while other methods can return reflections rather than true rotations as a degenerate solution [17].

**RANSAC parameters** We performed light tuning of the RANSAC parameters by considering only the teddybear category. Two parameters are important: the maximum number of trials, and the inlier threshold. As the point clouds in CO3D are only recovered up to a scale, the authors choose the convention of scaling them to have a unit standard deviation averaged across the three world axes. This makes the choice of a single inlier threshold to be used across all categories possible. In our experiments, we choose 0.2 as this threshold, which in the context of the rigid body transform solution means that any point that, under the recovered transform, is less than a 0.2 Euclidian distance away from its corresponding point, is considered an inlier.

The second important parameter for RANSAC is the number of trials that are run. We chose to limit this to keep inference to a few seconds, and use 1,000 trials for all categories. With 5 target views, this gives the 46.3% Acc30 reported in the main paper. Using 500 trials, this drops to 45.8%, and using 2,000 trials, it rises to 46.6%.

Finally, we sample 4 correspondences within every RANSAC trial to compute the rigid body transform. Solutions to this problem can suffer from degeneracy with only 3 points [17].

### B.5 Depth-free methods

| | Med. Err (↓) | Acc30 (↑) | Acc15 (↑) |
|---|---|---|---|
| Ours ($K = 50$, No Depth) | 81.7 | 23.6 | 6.8 |
| Ours-BV ($K = 50$, With Depth) | 61.1 | 35.4 | 10.6 |
| Ours ($K = 50$, With Depth) | **53.8** | **46.3** | **31.1** |

Fig. A.4: We investigate the performance of our method in a depth-free setting, finding that allowing access to depth substantially improves pose estimation performance. 'Ours-BV' indicates simply assuming that the best target view (recovered with our method) is perfectly aligned with the reference.

We permit depth maps in our setting as we believe them to be readily available in many practical scenarios, whether through SfM (as in our experiments with CO3D), depth cameras, or stereo. However, here we include a depth-free algorithm where, following 'best view' retrieval with our method, we estimate the essential matrix between this and the reference, and extract pose from this. We show the results in fig. A.4, finding that this depth-free method is not as robust as our depth-based method, for the fine-grained alignment task required after best view retrieval (though it still outperforms the baselines). Specifically, we find pose prediction through essential matrix estimation to be worse than simply predicting an identity transform on top of the best-view. We suggest further investigation is warranted in depth-free variants of our setting.

### B.6 Analysis of results

**Choice of evaluation metrics** It has long been noted that when reporting pose estimation errors and accuracies, results can be skewed by the presence of rotationally symmetric objects, where a 'reasonable' pose estimate can nonetheless be assigned a very high geodesic error (e.g. a toaster that is predicted to have an azimuth angle of 180º rather than 0º — both settings would have very similar appearance). For this reason, some works that assume access to object CAD models or point clouds relax the evaluation of pose estimation. For instance, [23] propose the closest point distance metric for symmetric objects, which naturally accounts for symmetries by summing the distances between all points on an object under the predicted pose, and the *closest* points to these on the reference object under the ground-truth pose.

In this work, we use accuracy (at 15º, 30º) and median error metrics, as is conventional across much of the pose estimation literature. Our reasons for this are twofold. Firstly, cross-instance shape gap makes closest point distance metrics, used in the single-instance literature to handle symmetry, ill-posed. A 'perfect' relative pose prediction between two object instances would nonetheless carry a non-zero distance due to shape differences. Second, the choice of whether or not to use the closest point distance is made based on whether an object has a rotational symmetry or not [23]. In the zero-shot setting, this cannot be known either a-priori or at test time. Our metrics are thus sensitive to symmetries, but the most appropriate choice for category-level pose estimation. To reduce the impact of symmetries in skewing the reported results, we do not consider object categories with infinite rotational symmetry (see appendix A.2).

**Impact of near rotational symmetry on results** Many of the 20 categories included in our evaluation exhibit *near* rotational symmetry between a 0º and 180º azimuthal view (about the gravitational axis). For instance, most instances in the *handbag* category have almost complete rotational and mirror symmetry in this sense, with labelling using cues from the handle position to disambiguate pose (see appendix A.2). To inspect the extent to which categories such as this affect our results, which as just discussed use metrics that enforce a single correct pose label, we plot geodesic errors in 3D orientation prediction from our method in fig. A.5. Inspect these results, it can be seen that categories that intuitively have a near-symmetry at a 180º offset do tend indeed exhibit a strong bi-modal prediction error that reflects this. For the *chair* and *toaster* categories, where some instances further have 90º rotational symmetry, a third mode of error can be seen that reflects this, also.

# C Baselines

## C.1 Iterative closest point

Iterative Closest Point (ICP) methods seek to minimise the distance between a reference and target point cloud, in the absence of known correspondences, by computing the optimal rigid body transform between these clouds, in an iterative manner. We use the implementation of ICP in the Pytorch3D library [34], and include a uniform scaling parameter, to match our method's setting. The time complexity of ICP in the number of points $n$ is $O(n^2)$, and in order to keep the run-time tractable, we sub-sample each object's point cloud at random to

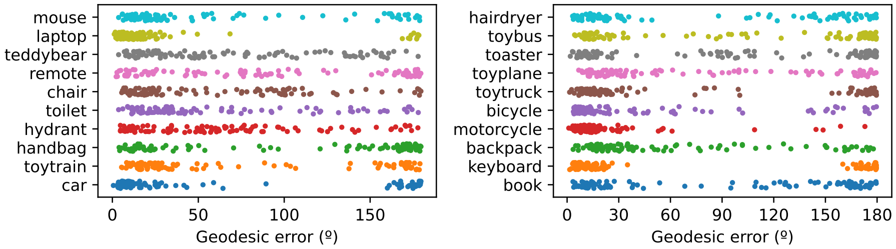

Fig. A.5: The results from 100 pose estimation problems for each of the 20 categories considered (for the 10 view setting considered in appendix B.1). A small amount of vertical displacement is added to the plotted points to make larger clusters salient. For many of the categories, a clear second mode is observed towards maximal geodesic error. In CO3D, where objects tend to vary mostly by an azimuthal rotation (about the gravitational axis), this often reflects a failure mode of predicting 'back-to-front' pose for objects that almost exhibit a rotational symmetry between the 0º and 180º azimuthal views (e.g. *bicycles, cars, keyboards, handbags*).

5000 points prior to running ICP. For the reference object, we construct a point cloud by back-projecting the single reference image using its depth map. For the target object, if multiple views are available, we leverage all of these for a more complete point cloud. We use the labelled foreground masks provided in CO3D to produce a masked point cloud - we do not use this in our method except to take a region of interest crop.

As discussed in section 5.2, we try running ICP both without any pose initialisation (**ICP**), and - in the multiple target view settings - with initialisation given by the predicted 'best frame' from our method. When running without initialisation, we first transform the point clouds to put them in a coordinate frame induced by assuming that the viewing camera (in the reference frame, or in the first frame of the target sequence) is in the position of the camera in first frame of the sequence. That is, for the $i^{\text{th}}$ reference frame $\text{ref}_i$, we transform the reference point cloud by $\mathbf{T}^{\text{cam}}_{\text{ref}_0} \circ (\mathbf{T}^{\text{cam}}_{\text{ref}_i})^{-1}$, where $\mathbf{T}^{\text{cam}}$ denotes a world-to-view camera transform, and $\text{ref}_0$ is the first frame in the reference sequence. This is to reduce a bias in CO3D towards point clouds that are very nearly already aligned in their standard coordinate frames - the camera extrinsic orientation is always the same in first frame of each sequence, and the point cloud coordinate frame is defined with respect to this. For most categories, the crowd-sourced videos start from a very similar viewpoint, which leads to nearly aligned point clouds. When initialising from a best-frame estimate with index $j^*$, we use this frame's extrinsics to transform the reference point cloud i.e. $\mathbf{T}^{\text{cam}}_{\text{ref}_0} \circ (\mathbf{T}^{\text{cam}}_{\text{ref}_{j*}})^{-1}$ to bring it in line with this view.

### C.2 PoseContrast

PoseContrast [49] is an RGB-based method designed for zero-shot category level 3D pose estimation. In contrast to our work, it only estimates SO(3) pose, with no translation estimate. It makes use of a pre-trained ResNet50 backbone, and trains on pose-labelled category-level datasets (Pascal3D [47] and Objectnet3D [46]) with a contrastive loss based on the geodesic difference in pose between samples. Intuitively, it seeks to learn an embedding space in which objects of similar pose are closer together, in the hope that this will generalise to previously unseen categories. The authors note that zero-shot successes are still only probable in cases in which the unseen category has both similar appearance, geometry and canonical reference frame to a category in the training set. As canonical reference frames can be arbitrarily chosen, this makes the success or otherwise of this method entirely dependent on a dataset's choice for category reference frames. In the present work, we formulate zero-shot pose as agnostic of canonical frame, by providing the reference frame implicitly through use of a single reference image. To directly compare to PoseContrast, we bring PoseContrast to the relative setting too. First, PoseContrast estimates a 3D pose for both reference and target frames individually. We then compute the relative SO(3) transform between these two estimates to form the final prediction. We then compare this to the ground-truth given by our labelling process as in all other methods.

Despite the presence of some of our considered categories (e.g. *toaster*) in the ObjectNet3D training set used by PoseContrast, we find that this method does not perform well in our setting. Inspecting the output predictions for individual categories, we find that for certain categories it appears to exploit the uneven viewpoint distributions in the ObjectNet3D dataset, rather than learning meaningful pose estimates.

## References

1. Aberman, K., Liao, J., Shi, M., Lischinski, D., Chen, B., Cohen-Or, D.: Neural best-buddies: Sparse cross-domain correspondence. ACM Transactions on Graphics (2018)
2. Ahmadyan, A., Zhang, L., Ablavatski, A., Wei, J., Grundmann, M.: Objectron: A Large Scale Dataset of Object-Centric Videos in theWild with Pose Annotations. In: CVPR (2021)
3. Akizuki, S.: ASM-Net : Category-level Pose and Shape Estimation Using Parametric Deformation. In: BMVC (2021)
4. Amir, S., Gandelsman, Y., Bagon, S., Dekel, T.: Deep ViT Features as Dense Visual Descriptors (2021)
5. Bertalmio, M., Bertozzi, A., Sapiro, G.: Navier-stokes, fluid dynamics, and image and video inpainting. In: CVPR (2001)
6. Caron, M., Misra, I., Mairal, J., Goyal, P., Bojanowski, P., Joulin, A.: Unsupervised learning of visual features by contrasting cluster assignments. NeurIPS (2020)
7. Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., Joulin, A.: Emerging Properties in Self-Supervised Vision Transformers. In: ICCV (2021)
8. Chen, D., Li, J., Wang, Z., Xu, K.: Learning canonical shape space for category-level 6D object pose and size estimation. In: CVPR (2020)
9. Chen, K., Dou, Q.: SGPA: Structure-Guided Prior Adaptation for Category-Level 6D Object Pose Estimation. In: ICCV (2021)
10. 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: CVPR (2021)
11. Chen, X., Fan, H., Girshick, R.B., He, K.: Improved baselines with momentum contrastive learning (2020), https://arxiv.org/abs/2003.04297
12. Chen, X., Dong, Z., Song, J., Geiger, A., Hilliges, O.: Category Level Object Pose Estimation via Neural Analysis-by-Synthesis. In: ECCV (2020)
13. Cheng, X., Lin, H., Wu, X., Yang, F., Shen, D.: Improving video-text retrieval by multi-stream corpus alignment and dual softmax loss (2021)
14. Choi, S., Zhou, Q.Y., Miller, S., Koltun, V.: A Large Dataset of Object Scans (2016)
15. Deng, X., Xiang, Y., Mousavian, A., Eppner, C., Bretl, T., Fox, D.: Self-supervised 6D Object Pose Estimation for Robot Manipulation. In: ICRA (2020)
16. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N.: An image is worth 16x16 words: Transformers for image recognition at scale. ICLR (2021)
17. Eggert, D.W., Lorusso, A., Fisher, R.B.: Estimating 3-D rigid body transformations: A comparison of four major algorithms. Machine Vision and Applications (5-6), 272–290 (1997)
18. El Banani, M., Corso, J.J., Fouhey, D.F.: Novel object viewpoint estimation through reconstruction alignment. In: CVPR
19. Florence, P.R., Manuelli, L., Tedrake, R.: Dense Object Nets: Learning Dense Visual Object Descriptors By and For Robotic Manipulation. In: CoRL (2018)
20. Goodwin, W., Vaze, S., Havoutis, I., Posner, I.: Semantically Grounded Object Matching for Robust Robotic Scene Rearrangement. In: ICRA (2021)
21. Grabner, A., Roth, P.M., Lepetit, V.: 3D Pose Estimation and 3D Model Retrieval for Objects in the Wild. In: CVPR (2018)

22. Gupta, S., Arbeláez, P., Girshick, R., Malik, J.: Inferring 3D Object Pose in RGB-D Images (2015)
23. 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: Lecture Notes in Computer Science (2013)
24. Huynh, D.Q.: Metrics for 3d rotations: Comparison and analysis. J. Math. Imaging Vis. (2009)
25. Kanezaki, A., Matsushita, Y., Nishida, Y.: RotationNet: Joint Object Categorization and Pose Estimation Using Multiviews from Unsupervised Viewpoints. In: CVPR (2018)
26. Kundu, J.N., Rahul, M.V., Ganeshan, A., Babu, R.V.: Object pose estimation from monocular image using multi-view keypoint correspondence. In: ECCV (2019)
27. Lee, J., Kim, D., Ponce, J., Ham, B.: SFNET: Learning object-aware semantic correspondence. In: CVPR (2019)
28. Li, X., Weng, Y., Yi, L., Guibas, L., Abbott, A.L., Song, S., Wang, H.: Leveraging SE(3) Equivariance for Self-Supervised Category-Level Object Pose Estimation. In: NeurIPS 2021 (2021)
29. Lin, Y., Tremblay, J., Tyree, S., Vela, P.A., Birchfield, S.: Single-stage Keypoint-based Category-level Object Pose Estimation from an RGB Image (2021)
30. Liu, Y., Zhu, L., Yamada, M., Yang, Y.: Semantic correspondence as an optimal transport problem. In: CVPR (2020)
31. Lowe, D.G.: Distinctive Image Features from Scale-Invariant Keypoints. International Journal of Computer Vision (2004)
32. Manuelli, L., Gao, W., Florence, P., Tedrake, R.: kPAM: KeyPoint Affordances for Category-Level Robotic Manipulation. In: International Symposium on Robotics Research (ISRR) (2019)
33. Pavlakos, G., Zhou, X., Chan, A., Derpanis, K.G., Daniilidis, K.: 6-DoF Object Pose from Semantic Keypoints. ICRA (2017)
34. Ravi, N., Reizenstein, J., Novotny, D., Gordon, T., Lo, W., Johnson, J., Gkioxari, G.: Accelerating 3d deep learning with pytorch3d (2020), https://arxiv.org/abs/2007.08501
35. Reizenstein, J., Shapovalov, R., Henzler, P., Sbordone, L., Labatut, P., Novotny, D.: Common Objects in 3D: Large-Scale Learning and Evaluation of Real-life 3D Category Reconstruction. In: ICCV (2021)
36. Sahin, C., Kim, T.K.: Category-level 6D object pose recovery in depth images. In: ECCV (2019)
37. Schonberger, J.L., Frahm, J.M.: Structure-from-Motion Revisited. In: CVPR (2016)
38. Shi, J., Yang, H., Carlone, L.: Optimal Pose and Shape Estimation for Category-level 3D Object Perception. In: Robotics: Science and Systems XVII (2021)
39. Simeonov, A., Du, Y., Tagliasacchi, A., Tenenbaum, J.B., Rodriguez, A., Agrawal, P., Sitzmann, V.: Neural Descriptor Fields: SE(3)-Equivariant Object Representations for Manipulation (2021)
40. Tian, M., Ang, M.H., Lee, G.H.: Shape Prior Deformation for Categorical 6D Object Pose and Size Estimation. In: ECCV (2020)
41. Tseng, H.Y., De Mello, S., Tremblay, J., Liu, S., Birchfield, S., Yang, M.H., Kautz, J.: Few-shot viewpoint estimation. In: BMVC (2020)
42. Umeyama, S.: Least-Squares Estimation of Transformation Parameters Between Two Point Patterns. IEEE Transactions on Pattern Analysis and Machine Intelligence (1991)

43. Vaze, S., Han, K., Vedaldi, A., Zisserman, A.: Generalized category discovery. CVPR (2022)
44. Wang, A., Kortylewski, A., Yuille, A.: NeMo: Neural Mesh Models of Contrastive Features for Robust 3D Pose Estimation. In: ICLR (2021)
45. Wang, H., Sridhar, S., Huang, J., Valentin, J., Song, S., Guibas, L.: Normalized Object Coordinate Space for Category-Level 6D Object Pose and Size Estimation. CVPR (2019)
46. Xiang, Y., Kim, W., Chen, W., Ji, J., Choy, C., Su, H., Mottaghi, R., Guibas, L., Savarese, S.: Objectnet3D: A large scale database for 3D object recognition. In: ECCV (2016)
47. Xiang, Y., Mottaghi, R., Savarese, S.: Beyond PASCAL: A benchmark for 3D object detection in the wild. In: WACV (2014)
48. Xiang, Y., Schmidt, T., Narayanan, V., Fox, D.: PoseCNN: A Convolutional Neural Network for 6D Object Pose Estimation in Cluttered Scenes. In: Robotics: Science and Systems XIV (2018)
49. Xiao, Y., Du, Y., Marlet, R.: PoseContrast: Class-Agnostic Object Viewpoint Estimation in the Wild with Pose-Aware Contrastive Learning. In: 3DV (2021)
50. Xiao, Y., Marlet, R.: Few-Shot Object Detection and Viewpoint Estimation for Objects in the Wild. In: ECCV (2020)
51. Xiao, Y., Qiu, X., Langlois, P.A., Aubry, M., Marlet, R.: Pose from Shape: Deep pose estimation for arbitrary 3D objects. In: BMVC (2019)
52. Zhou, Q., Park, J., Koltun, V.: Open3d: A modern library for 3d data processing (2018), http://arxiv.org/abs/1801.09847
53. Zhou, X., Karpur, A., Luo, L., Huang, Q.: StarMap for Category-Agnostic Keypoint and Viewpoint Estimation. In: ECCV (2018)