Templates for 3D Object Pose Estimation Revisited: Generalization to New Objects and Robustness to Occlusions

Van Nguyen Nguyen[1], Yinlin Hu[2], Yang Xiao[1], Mathieu Salzmann[2], Vincent Lepetit[1]
[1]LIGM, Ecole des Ponts, Univ Gustave Eiffel, CNRS, France
[2]CVLab, EPFL, Switzerland
{van-nguyen.nguyen, yang.xiao, vincent.lepetit}@enpc.fr
{yinlin.hu, mathieu.salzmann}@epfl.ch

Abstract

We present a method that can recognize new objects and estimate their 3D pose in RGB images even under partial occlusions. Our method requires neither a training phase on these objects nor real images depicting them, only their CAD models. It relies on a small set of training objects to learn local object representations, which allow us to locally match the input image to a set of "templates", rendered images of the CAD models for the new objects. In contrast with the state-of-the-art methods, the new objects on which our method is applied can be very different from the training objects. As a result, we are the first to show generalization without retraining on the LINEMOD and Occlusion-LINEMOD datasets. Our analysis of the failure modes of previous template-based approaches further confirms the benefits of local features for template matching. We outperform the state-of-the-art template matching methods on the LINEMOD, Occlusion-LINEMOD and T-LESS datasets. Our source code and data are publicly available at https://github.com/nv-nguyen/template-pose.

1. Introduction

3D object pose estimation has significantly improved over the past decade in terms of both robustness and accuracy [17, 29, 33, 19, 43]. In particular, the robustness to partial occlusions has greatly increased [27, 16, 23], and the need for large amounts of real annotated training images has been relaxed thanks to domain transfer [1], domain randomization [35, 18, 30], and self-supervised learning [32] techniques that leverage synthetic images for training.

Nevertheless, the use of image-based 3D object pose estimation remains limited in the industry, despite its huge potential for robotics and augmented reality. Scalable industrial applications would, for example, require the ability to handle arbitrary, previously-unseen objects without retraining and with access only to the objects' CAD models, thus saving both training and data capture time. While a few works have already tackled this challenging task [30, 28, 38, 2], most of them impose some additional constraints by assuming that the new objects belong to a known category [37], remain similar to the training ones as in the T-LESS dataset [30], or have prominent corners [28].

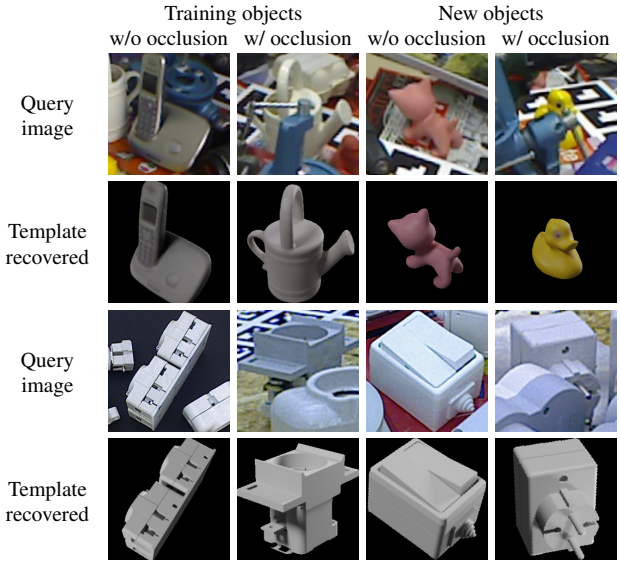

Figure 1: Our method can estimate the 3D pose of new objects in query images by matching them with templates created from their 3D models. **These new objects can be very different from the ones, and can be partially occluded in the query images.**

By contrast, template-based approaches [38, 2] offer the promise of generalizing to arbitrary new objects by learning an image embedding used to match the input image to a series of templates generated from their CAD models. Unfortunately, their use with new objects has been demonstrated only anecdotally, and we show in our experiments that these methods struggle in this challenging scenario, particularly in the presence of occlusions. We indeed notice that the

global representations used in [38, 2] to compare the input image to the CAD-generated templates have two limitations. First, they generalize poorly to new objects in the presence of a cluttered background, and result in inaccurate pose estimation even for uniform background. Furthermore, they are ill-suited to handle occlusions.

These observations motivate us to keep the 2D structure of the images for a template-based approach. More precisely, given a small set of training objects, we learn local features that can be used to reliably match real images and synthetical templates. Relying on local features allows us to discard the background: While the object's mask in the input image is not available at run-time, we can use the template's mask, thus solving the first limitation of global representations. Note that using the template's mask to instead remove the background in the real image before computing the image global representation requires us to recompute the input image representation for each template, which would result in very slow matching.

As will be shown by our experiments, using local features also results in much more accurate poses. This can be explained by the fact that we do not use pooling operations, which remove critical information about the poses, especially for new objects. Finally, yet another advantage is that our method can be robust to partial occlusions. To do so, we introduce a measure to evaluate the similarity between two images that explicitly takes into account the object's mask in the template and the possible occlusions in the query image.

We demonstrate the benefits of our approach on the LINEMOD [11], Occlusion-LINEMOD [3], and T-LESS [13] datasets. It consistently outperforms previous works [38, 2, 31, 30] on new objects by a large margin. In summary, our contributions are:

- A failure-case analysis of previous template-based methods when testing on new objects;
- A method that can predict the pose of new objects from their CAD models, without training on these objects nor restricting these objects to be similar to the training ones;
- A method robust to occlusions even in the challenging scenario when objects are both new and occluded.

2. Related Work

Our goal is to develop a method able to estimate the 3D pose of previously-unseen objects while having access only to their 3D model. It should be noted that early approaches to 3D pose estimation already targeted this goal [21]. However, these approaches, based on image edges and object contours, proved to be very fragile. As discussed below, with the use of deep learning, methods have become much more robust but typically require many training images.

Pose estimation for known objects. Many 3D object pose estimation methods use a deep model trained on real images or synthetic renderings of these objects, [17, 29, 19, 33, 20, 43, 25, 15]. Some also show remarkable robustness to partial occlusions of the objects [23, 27, 16]. Such an approach however requires long expensive training and data acquisition/generation time, which we would like to avoid. For example, the state-of-the-art method [18] on standard benchmarks [14] requires almost a day on 32 GPUs for training. While some works have attempted to reduce the burden of registering real images by learning to generate new images from real ones [26], their cost remains too cumbersome for many practical applications.

Category-level pose estimation. One way to avoid retraining on new object instances is to consider object categories, and train a model on target categories that will generalize to new instances of these categories [44, 37]. While such an approach can be useful in some applications, such as scene understanding, in many others, the new objects do not belong to a known category. By contrast, our approach generalizes to new objects that bear no similarity in shape with the known objects used to train the initial model.

Unseen object pose estimation. [38] proposed to learn discriminative representations of templates, which are images of objects associated with the corresponding 3D poses. Pose estimation could then be achieved by matching the input image against these templates in an image-retrieval manner. In this context, [2] then showed how to obtain more discriminative representations. While the ability to consider unseen objects by using their 3D models seems to be the motivation for these works, this was only superficially demonstrated, and our experiments show that these methods perform poorly on unseen objects.

More recently, [30] proposed an extension of [31] to generalize to unseen objects. This method introduces a novel architecture with multiple decoders to adapt to different object types. While their results indeed show generalization to unseen objects, these objects must remain similar to the training ones. As a consequence, this method has been demonstrated only on the T-LESS dataset, which depicts different kinds of electrical appliances that bear strong visual similarities.

In any event, as we will discuss in detail in Section 3.1, these methods rely on a global representation of the templates. We will show that our local representation-based framework has significant advantages in terms of generalization to new objects and of robustness to occlusions.

[28] also considers local representations but in a way that is very different from us: [28] learns to detect specific 2D object locations in the image together with a descriptor for each such location to match them with 3D points on the object's 3D model. This matching, however, is done indepen-

dently for each location, making it highly ambiguous, and resulting in a combinatorial matching cost and frequent failures. By contrast, we extract local representations in a grid structure and learn to match all local input and template representations jointly. To achieve this, we rely on contrastive learning, which we discuss below.

A different and interesting take was proposed in [42], where the embedding of the object's 3D model was used as input in addition to the input image to predict the 3D pose. However, this work considers only pose regression and assumes the object is already known in order to use the right 3D model.

Contrastive learning. Given a collection of images, contrastive learning aims to learn an embedding space where similar images are close to each other while dissimilar ones are far apart. [12, 39, 24, 34, 9, 5] leverage unlabeled images and strong data augmentation to learn powerful image features that achieve results competitive with those of supervised learning on various downstream tasks.

In our context, [41] exploits a form of contrastive learning, leveraging the pose labels to learn a pose-aware embedding space for class-agnostic 3D object pose estimation. One limitation of [41] is that different objects can be mixed with each other in the embedding space, thus making it impossible to recognize the correct object instance from the input image. Moreover, like [42], [41] does not attempt to recognize the object.

By contrast, [38, 2] rely on contrastive learning to learn an embedding space that is variant to both the object pose and the object instance. To this end, they rely on a triplet loss for learning object-discriminative features, together with a pairwise loss for pose-discriminative features. Similarly, we use contrastive learning to extract a discriminative feature representation, but we show that the InfoNCE [24] loss is the most simple and effective choice. Our experiments also show that most of the performance of our method in terms of generalization and robustness to occlusions come from our use of local representations.

3. Method

Our goal is to recognize new objects in color images and predict their 3D poses. We do this by matching the color image of the object with a set of templates. A template is a rendered image of a 3D model in some 3D pose. For each new object, the template set contains many templates, rendered from different views sampled around its 3D model. As the templates are annotated with the object's identity and pose, the method returns the identity and pose of the template most similar to the input image.

The challenge then is to measure the similarity between templates and input images. This should be done reliably despite that no real images of the new objects have been seen beforehand, the objects can be partially occluded, the lighting differs between the templates and the real images, and the object's background is cluttered in the real images.

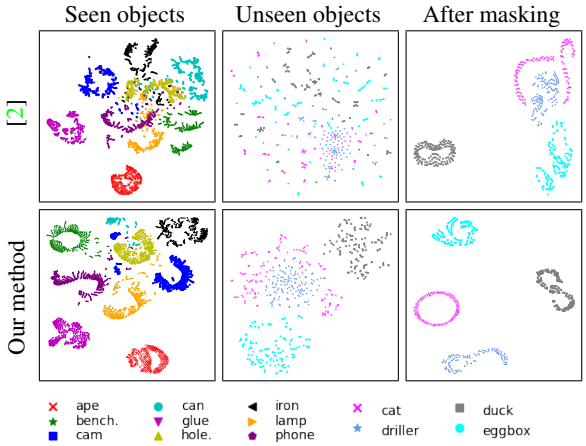

Figure 2: **Understanding the influence of background on different image representations,** with T-SNE visualizations of the image representations learned by [2] (first row) and by our method (second row) for real images of LINEMOD objects. For a given column, all the plots have the same scale for comparison.

In this work, motivated by the better repeatability and robustness to occlusions of local representations compared to global ones, we measure the similarity between an input image and a template based on local image features extracted using a deep model. We train this model using pairs made of a real image and a synthetic image from a small set of training objects. Note that these training objects can be very different in appearance from the new objects.

We start this section with an analysis of the limits of global representations in Section 3.1. We then detail in Section 3.2 our training procedure. It relies on a similarity measure that compares the local features of real images and synthetic templates. At run-time, we use an extended version of this similarity function that explicitly estimates which local features in the input image are occluded and discards them. We discuss this in Section 3.3. Finally, we detail how we generate the templates in Section 3.4.

3.1. Motivation and Analysis

Here, we present two experiments that point out the main drawbacks of global representations in template matching when working with unseen objects.

3.1.1 Cluttered Background

A first drawback of global representations is their poor ability to represent unseen objects on cluttered backgrounds. To show this, we plot in Figure 2 the t-SNE visualization [36] of the global representations learned by [2] and local representations learned by our method for real images of training and new objects of the LINEMOD dataset.

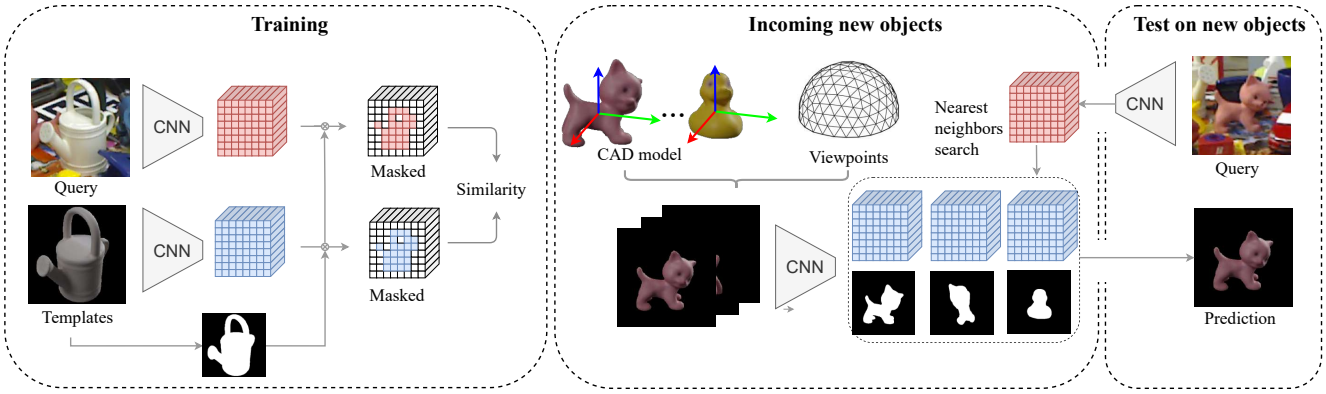

Figure 3: **At training time**, we use pairs made of a real image and a synthetic template to train a network to compute local features, from which the similarity between the two images can be predicted. **At run-time**, we apply this network to images of objects not seen during training to compute their local features. We can then retrieve the object pose by matching the image against the database of templates.

The first column of Figure 2 shows that both representations manage to cluster the images of each training object together, despite the fact that the images of the objects are captured with a cluttered background. The second column shows that global representations of [2] cannot disentangle the images of unseen objects, while our representations can. To better understand the reason behind this, we remove the background in the images by replacing it with a uniform color using the ground-truth object masks. As shown in the third column, the representations are now disentangled. This shows the influence of the background on the global representations for unseen objects, and that our representations are robust to cluttered backgrounds.

3.1.2 Pose Discrimination

A second drawback of global representations is their poor reliability when matching the real image of an unseen object with the synthetic template for the corresponding 3D pose, even when the object identity is known and the background is uniform. This can be explained by the fact that the pooling layers remove important information. This information loss appears to be compensated by the rest of the architecture for the training objects, but this compensation does not generalize to unseen objects.

To show this, we visualize in the supplementary material the correlation between pose distances and representation distances for unseen objects, as done in [38, 2]. While both representations result in a strong correlation for training objects, this correlation is lost when considering unseen objects for the global representations but not for ours. Even without background, the correlation is still very low for global representations [2].

3.2. Framework

In each training iteration, we sample N positive pairs, where pair i is composed of a real image $\mathbf{q}_i$ depicting a training object and of a synthetic template $\mathbf{t}_i$ of the same object in a similar 3D pose. Following [38], we deem the two viewpoints similar if the angle between them is less than 5 degrees. All the pairs composed by a real image and a synthetic image of different objects or dissimilar poses (larger than 5 degrees) are defined as negative pairs.

Triplet loss. [38] proposed a metric learning approach based on the intuition that the distance between feature descriptors for positive pairs should be closer in the learnt embedding space than negative pairs. To learn this property, [38] used a training loss $\mathcal{L} = \mathcal{L}_{triplet} + \mathcal{L}_{pair}$ where:

- $\mathcal{L}_{triplet}$ is the triplet term, which allows the network to learn features such that the distance in the learned embedding space between the positive pairs $\Delta_+^{(i)}$ is lower than the distance between the negative pairs $\Delta_-^{(i)}$ within the limits of the margin m. This triplet term is defined as

$$\mathcal{L}_{triplet} = \sum_{i=1}^{N} \max\left(0, 1 - \frac{\Delta_+^{(i)}}{\Delta_-^{(i)} + m}\right) \quad (1)$$

- $\mathcal{L}_{pair} = \sum_{i=1}^{N} \Delta_+^{(i)}$ is the pairwise term, to minimise distances between two images of identical poses but different viewing conditions.

[2] made an extension of this work by proposing a triplet loss which focuses only on learning object-discriminative features while using a pairwise loss to learn an embedding space analogous to the pose differences.

While these two losses work well, we experimentally show that the recent standard contrast loss InfoNCE [24] is the most simple and effective choice.

InfoNCE loss. For each real image $\mathbf{q}_i$, we also create $N-1$ negative pairs by combining it with synthetic templates $\mathbf{t}_k$ of other pairs in the current batch, with $1 \leq k \leq$

$N, k \neq i$. Altogether, this yields N positive pairs and $(N-1) \times N$ negative pairs for each batch. We train our model to maximize the agreement between the representations of samples in positive pairs, while minimizing that of negative pairs with the InfoNCE loss function [24]:

$$\mathcal{L} = -\sum_{i=1}^{N} \log \frac{\exp\left(\text{sim}(\overline{\mathbf{q}}_i, \overline{\mathbf{t}}_i)/\tau\right)}{\sum_{k=1}^{N} 1_{[k \neq i]} \exp\left(\text{sim}(\overline{\mathbf{q}}_i, \overline{\mathbf{t}}_k)/\tau\right)}, \quad (2)$$

where $\text{sim}(\overline{\mathbf{q}}, \overline{\mathbf{t}})$ measures the similarity between the local image features $\overline{\mathbf{q}}$ and $\overline{\mathbf{t}}$ computed by the deep model for real image $\mathbf{q}$ and template $\mathbf{t}$, and $\tau = 0.1$, is a temperature parameter. As shown in Figure 3, $\overline{\mathbf{q}}$ and $\overline{\mathbf{t}}$ retain a grid structure and are 3-tensors. In practice, their dimensions depend on the size of the input image, ranging from $25 \times 25 \times C$ to $28 \times 28 \times C$, with $C = 16$.

Local feature similarity. While previous works on contrastive learning [24, 34, 22, 4, 9, 7, 5, 6] focused mostly on image classification and define the similarity metric $\text{sim}(.,.)$ using a global representation of the two images, we found such a representation to only classify well either known objects or images with a clean background, as discussed in Section 3.1.1. To effectively handle new objects and complex backgrounds, we use a metric based on a pairwise comparison of the local features in $\overline{\mathbf{q}}$ and $\overline{\mathbf{t}}$. Specifically, we define

$$\text{sim}(\overline{\mathbf{q}}, \overline{\mathbf{t}}) = \frac{1}{|\mathcal{M}|} \sum_{l} \mathcal{M}^{(l)} \mathcal{S}\left(\overline{\mathbf{q}}^{(l)}, \overline{\mathbf{t}}^{(l)}\right), \quad (3)$$

where $\mathcal{S}$ is a local similarity metric, $\mathcal{M}$ is a 2D binary visibility mask for template $\mathbf{t}$, and index l indicates a 2D grid location. $\overline{\mathbf{q}}^{(l)}$ and $\overline{\mathbf{t}}^{(l)}$ are thus local features of dimension C. Considering the template mask allows us to discard the background in the real image. Note that the mask does not account for possible occlusions in the real image as it corresponds to the object's silhouette in the template. Occlusions will be considered in the next subsection. As a local similarity metric $\mathcal{S}$, we use the cosine similarity

$$\mathcal{S}\left(\overline{\mathbf{q}}^{(l)}, \overline{\mathbf{t}}^{(l)}\right) = \frac{\overline{\mathbf{q}}^{(l)}}{||\overline{\mathbf{q}}^{(l)}||} \cdot \frac{\overline{\mathbf{t}}^{(l)}}{||\overline{\mathbf{t}}^{(l)}||}, \quad (4)$$

We empirically observed that measuring the similarity as the opposite of the L1 and L2 norms of the differences yields the same performance as the cosine similarity.

3.3. Run-time and Robustness to Occlusions

At run-time, given a real query image $\mathbf{q}$, we retrieve the most similar template in a template set. To be robust to occlusions that can occur in the query image, we modify $\text{sim}(\overline{\mathbf{q}}, \overline{\mathbf{t}})$ as:

$$\text{sim}^*(\overline{\mathbf{q}}, \overline{\mathbf{t}}) = \frac{1}{|\mathcal{M}|} \sum_{l} \mathcal{M}^{(l)} \mathcal{O}^{(l)} \mathcal{S}\left(\overline{\mathbf{q}}^{(l)}, \overline{\mathbf{t}}^{(l)}\right), \quad (5)$$

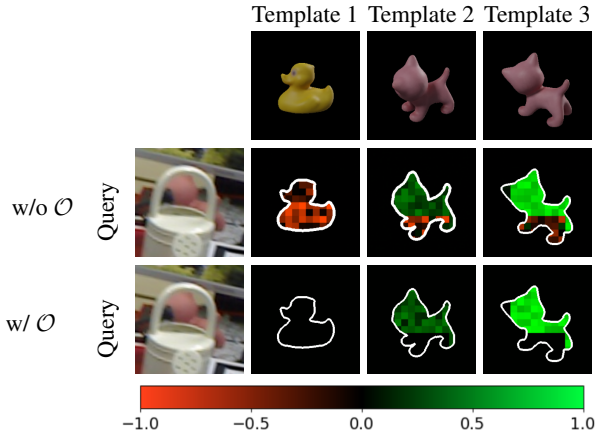

Figure 4: **Illustration of feature similarity** when not using the occlusion mask $\mathcal{O}$ (second row) and when using it. As discussed in Section 3.3, using $\mathcal{O}$ allows "turning off" the possible occluded local features in the similarity score.

where $\mathcal{O}^{(l)} = 1_{\mathcal{S}(\overline{\mathbf{q}}^{(l)}, \overline{\mathbf{t}}^{(l)}) > \delta}$ with δ a threshold applied to the cosine similarity to "turn off" the occluded local features as shown in Figure 4. In practice, we set this threshold $\delta = 0.2$ through ablation study. Note that Eq. (5) can be written as the element-wise product $\odot$ and can be computed efficiently with:

$$\text{sim}^*(\overline{\mathbf{q}}, \overline{\mathbf{t}}) = \frac{1}{|\mathcal{M}|} (\mathcal{M} \odot \mathcal{O} \odot \mathcal{S}). \quad (6)$$

3.4. Template Creation

On LINEMOD [11] and Occlusion-LINEMOD [3] datasets, we follow the protocol of [38] to sample the synthetic templates. More precisely, the viewpoints are defined by starting with a regular icosahedron and recursively subdividing each triangle into 4 smaller triangles. After applying this subdivision two times and removing the lower half-sphere, we end up with 301 templates per object.

On T-LESS [13], we follow the protocol of [30] by using a dense regular icosahedron with 2'536 viewpoints and 36 in-plane rotations for each rendered image. Altogether, this yields 92'232 templates per object. Besides, we also show our results with a coarser regular icosahedron with 602 viewpoints, which results 21'672 templates per object.

We use BlenderProc [8] to generate templates with realistic rendering for both settings.

4. Experiments

In this section, we first describe the experimental setup (Section 4.1). Then, we compare quantitatively and qualitatively our method with previous works [38, 2, 31, 30] on both training (or seen) and unseen objects of the LINEMOD (LM) [11], Occlusion-LINEMOD (O-LM) [3] and T-LESS [13] datasets (Section 4.2). Finally, we provide

Split	Training	Seen LM	Seen O-LM	Unseen LM	Unseen O-LM
#1	9'954	981	6'832	4'848	2'377
#2	9'928	981	4'490	4'874	4'719
#3	8'850	872	7'096	6'061	2'113

Table 1: **Dataset splits for LM and O-LM.** For each split, we provide the numbers of real images in the training set and in four test sets.

an ablation study for investigating the effectiveness of our method with different parameters and failure cases of our method (Sections 4.3 and 4.4).

4.1. Experimental Setup

Data processing. For the LM and O-LM datasets, as there are no standard splits to evaluate the robustness of RGB-based methods on unseen objects, we propose three different splits created from the order of the object ids. The new, or unseen objects for each of these splits are:

- Split #1: **Ape**, Benchvise, Camera, **Can**;
- Split #2: **Cat**, **Driller**, **Duck**, **Eggbox**;
- Split #3: **Glue**, **Holepuncher**, Iron, Lamp, Phone.

The other objects from LM are used for training the model. The objects with names in **bold** in the lists above often are occluded in O-LM. Note that O-LM is only used for testing, as we do not need to see occlusions during the training time. Moreover, to understand the performance gap between objects that are seen or unseen during the training, we also evaluate the methods on seen objects. To do so, we keep 10% of the real images of training objects under unseen poses for testing purposes. Table 1 details the different splits.

On T-LESS [13], we follow the evaluation protocol of [30] by training only on objects 1-18 under randomized backgrounds of SUN397 [40] and testing on the complete T-LESS primesense test set. More details about training set of T-LESS can be found in the supplementary material.

Evaluation metrics. For the LM and O-LM datasets, the pose error is measured by the angle between the two positions on the viewing half-sphere. We also treat the "Eggbox" and "Glue" objects as symmetric around the z-axis as done in [38, 2].

In the case of known object pose estimation, the recognition score is almost 100% on LM and O-LM. Previous works [38, 2] that focused on known objects thus only evaluate the pose error without considering whether the retrieved object is actually correct. In the case of unseen objects, we found that retrieving correctly both pose and class is important as the model can still get correct poses but from another object. Therefore, we propose using the Acc15 metric, which measures how often the pose error is less than 15 degrees *and* the predicted object class is correct. We also report the pose error in the supplementary material.

As most objects in T-LESS [13] are symmetric, we report the recall under the Visible Surface Discrepancy (err_{vsd}) metric at $\text{err}_{vsd} < 0.3$ with tolerance $\tau = 20mm$ and $> 10\%$ object visibility as done in [31, 30]. Unless otherwise stated in previous works [31, 30], only templates of the same object are used at testing time (in other words, the class of the object is assumed to be known before testing). Please note that for the evaluation on the T-LESS dataset, we also predict the translation by using the same formula "projective distance estimation" of SSD-6D [17] as done in [31, 30]. This translation is deduced from the retrieved template and the input bounding box of query image. More details can be found in the supplementary material.

Implementation details. For a fair comparison, in the evaluation on LM and O-LM, we consider two different backbones: (i) "Base" – the simple backbone used in [38, 2]; (ii) ResNet50 – the standard backbone used in recent contrastive learning methods [9]. We reimplemented [38, 2] to get quantitative results in both seen and unseen objects. Our implementations get very similar performance when evaluated on the same data as the original papers on seen objects (see Table 2), validating our reimplementation.

We also follow [38, 2] when testing with the "Base" backbone by using the same input image of size 64×64. While testing with ResNet50, we use a larger input size of 224×224. In both settings, we slightly change the architecture by removing all the pooling, FC layers and then replace them by two 1×1 convolution layers to output the desired local feature of size 16. As done in [38, 2], we use the ground-truth pose to crop the input image at the center of objects and do not consider in-plane rotation (more details can be found in the supplementary material). On the T-LESS dataset, we use the same backbone ResNet50 and crop the input image with ground-truth bounding box as done in [31, 30].

For both evaluations, we train our networks using Adam with an initial learning rate of 1e-2 for the "Base" backbone and of 1e-4 for ResNet50. Training takes less than 5h for all splits on a single V100 GPU when training on LM [11] and around 12h when training on T-LESS [13].

4.2. Comparison with the State of the Art

4.2.1 LINEMOD and Occluded-LINEMOD Results

Table 2 presents the results of our method compared with previous work [38, 2]. With either the "Base" or ResNet50 backbones, our method based on local feature similarities achieves the best overall performance in almost all settings compared to previous methods that compute the feature similarity between global image representations. While [38, 2] explored carefully designed pairwise and triplet losses for learning an embedding space that is both object-discriminative and pose-discriminative, we find that using

Method	Backbone	Features	Loss	Seen LM				Seen O-LM				Unseen LM				Unseen O-LM			
				#1	#2	#3	Avg.	#1	#2	#3	Avg.	#1	#2	#3	Avg.	#1	#2	#3	Avg.
[38]	Base [38]	Global	[38]	87.0	83.1	85.1	85.0	19.2	23.1	15.0	19.1	13.2	15.5	18.2	15.2	9.3	5.1	5.1	6.5
[38]	Base [38]	Global	Eq. (2)	95.2	95.3	95.4	95.3	19.6	25.3	16.1	20.3	13.3	17.0	20.5	16.9	8.2	6.4	6.7	7.1
[2]	Base [38]	Global	[2]	89.2	85.4	83.3	86.3	18.3	21.9	17.6	19.5	14.1	16.3	19.7	16.7	8.2	7.5	7.6	7.8
[2]	Base [38]	Global	Eq. (2)	96.3	95.2	96.5	96.0	18.3	23.1	15.8	19.1	11.5	17.7	17.2	15.5	7.1	6.5	6.5	6.7
Ours	Base [38]	Local	[38]	84.8	85.5	86.3	85.5	50.1	51.3	42.2	47.9	69.6	63.2	46.2	59.7	35.3	34.3	44.2	37.9
Ours	Base [38]	Local	Eq. (2)	95.6	96.9	92.0	94.8	68.9	71.0	57.7	65.8	78.8	82.5	64.1	75.1	42.2	57.1	59.8	53.0
[38]	ResNet50 [10]	Global	Eq. (2)	98.8	96.9	98.8	98.1	66.7	73.2	62.7	67.5	42.2	43.7	49.4	45.1	22.3	22.5	45.9	29.9
[2]	ResNet50 [10]	Global	Eq. (2)	96.9	97.1	94.5	96.1	63.6	71.8	58.9	64.7	39.9	44.9	48.3	44.3	15.5	21.8	50.2	29.1
Ours	ResNet50 [10]	Local	Eq. (2)	**99.3**	**99.0**	**99.2**	**99.1**	**77.3**	**84.1**	**76.8**	**79.4**	**94.4**	**97.4**	**88.7**	**93.5**	**71.4**	**72.7**	**85.3**	**76.3**

Table 2: **Comparison of our method with [38] and [2]** on seen and unseen objects of LM and O-LM under the three different splits detailed at the beginning of Section 4.1. We report Acc15 ↑, the accuracy of predicting correctly the object identity *and* its pose with an error less than 15 degrees. We are on par on the "easy" case and outperform them by a large margin on the 3 other configurations. Using the InfoNCE loss rather than the loss from [2] brings some improvement, but the main improvement comes from our approach based on local features.

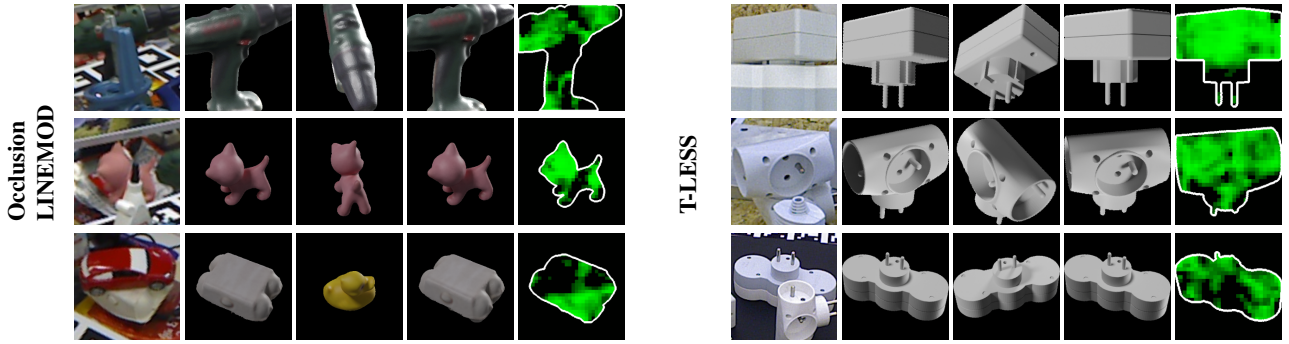

Figure 5: **Qualitative results on unseen objects** of Occlusion-LINEMOD (left) and T-LESS (right). Our method retrieves the correct template and pose while [2, 30] fails on unseen objects, particularly in the presence of occlusion.

Method	Number templates	Recall VSD		
		Obj. 1-18	Obj. 19-30	Avg
Implicit [31]	92K	35.60	42.45	38.34
MPL [30]	92K	35.25	33.17	34.42
Ours	92K	**59.62**	**57.75**	**58.87**
Ours	21K	59.14	56.91	58.25

Table 3: **Comparison with [31, 30]** on seen objects (obj. 1-18) and unseen objects (obj. 19-31) of T-LESS using the protocol from [30]. Our method significantly outperforms [31, 30] in the same setting.

the InfoNCE loss as defined in Eq. (2) boosts the performance of all methods, in particular for our method based on local feature similarities.

When the objects are occluded, the accuracy of [38, 2] drops to below 70% for training objects, while our method can still maintain a relatively high accuracy. This shows the robustness of local image features rather than global image representations that are much more strongly affected by the occlusions. Furthermore, the prediction accuracy of our method on unseen objects is clearly higher than that of previous methods, regardless of the objects being occluded or not. This indicates that matching based on local features is not only robust to occlusions, but also generalizes better to unseen objects. More importantly, this improvement on unseen objects holds still in the presence of occlusions.

4.2.2 T-Less Results

In Table 3, we shown that our proposed approach outperforms the state-of-the-art methods [31, 30] on the T-LESS dataset by a large margin on both seen and unseen objects. While [30] carefully designed single-encoder-multi-decoder network that allows sharing a latent space for all objects and having each decoder only reconstructs views of a single object, we find that using our method and InfoNCE loss is much more simple but also boosts significantly the performance in the same setting.

4.3. Ablation Study

We present several ablation evaluations on LINEMOD and Occlusion-LINEMOD.

	Ape	Can	Cat	Driller	Duck	Egg*	Glue*	Hole.	Avg
[38]	16.6	28.0	1.5	8.2	11.5	68.8	67.7	22.1	29.9
[2]	12.6	18.4	9.0	16.7	7.8	53.7	60.3	40.1	29.1
Ours	**53.8**	**89.7**	**45.1**	**84.4**	**87.2**	**76.9**	**89.9**	**83.3**	**76.3**
w/o $\mathcal{M}$	13.3	1.0	10.0	1.0	80.1	7.0	80.0	1.0	24.1

Table 4: **Effectiveness of $\mathcal{M}$.** Comparison of [38, 2] and our method with and without using the template mask $\mathcal{M}$ in the computation of the similarity. Using $\mathcal{M}$ allows discarding the cluttered background and brings significant improvement on occluded unseen objects.

Threshold δ	-0.3	-0.2	-0.1	0	0.1	0.2	0.3	w/o $\mathcal{O}$
Ape	54.1	53.7	54.6	**54.7**	54.0	53.8	53.6	53.3
Can	82.2	89.2	89.1	89.7	89.4	89.7	**89.8**	84.9
Cat	46.7	**47.5**	46.1	45.5	46.1	45.1	46.5	45.1
Driller	83.6	**84.5**	**84.5**	83.8	84.4	84.4	**84.5**	81.5
Duck	87.1	87.1	**87.8**	86.7	87.3	87.2	87.0	87.3
Egg*	76.3	75.2	74.1	75.3	75.1	**76.9**	76.2	72.6
Glue*	89.3	83.5	83.9	90.1	89.5	89.9	89.6	**90.2**
Holep.	83.9	**85.9**	83.6	82.9	83.4	83.3	82.5	81.8
Avg	75.4	75.8	75.4	76.0	76.1	**76.3**	76.2	74.5

Table 5: **Influence of threshold δ of Eq. (5).** Predicting occlusion mask $\mathcal{O}$ with threshold $\delta = 0.2$ results on the best performance, particularly on large objects.

Dataset	Number templates	Features creation	Memory	Run-time	
				CPU	GPU
LINEMOD	1.204	0.5 min	28 MB	0.15 s	7.8×10^{-3} s
T-LESS	21.672	6 min	544 MB	0.84 s	8.2×10^{-3} s

Table 6: **Average run-time** of our method on a single GPU V100 and CPU Intel Xeon.

Effectiveness of feature masking. Table 4 shows the effectiveness of using the template masks $\mathcal{M}$ in Eq. (6) for unseen objects. Removing $\mathcal{M}$ results in a dramatic degradation for our method on all the three splits.

Influence of the threshold δ. Table 5 shows the influence of the threshold δ in Eq. (5) for estimating the occlusion mask $\mathcal{O}$. Using $\mathcal{O}$ brings improvements on large objects ("Can", "Driller", and "Eggbox"). This can be explained by the fact that the occlusions can be very large in O-LM, especially on small objects, as shown in Figure 7.

Influence of the local feature dimensions. Figure 6 shows the pose error as a function of the dimension C of the local features and of the resolution of the feature maps and masks $\mathcal{M}$. While C is not a critical value, the resolution is more important, as higher resolution allows discarding the background more precisely. Furthermore, this hyperparameter has a much stronger influence on the performance on the unseen objects compared to the seen objects.

Run-time. Table 6 provides run-times on CPU and GPU.

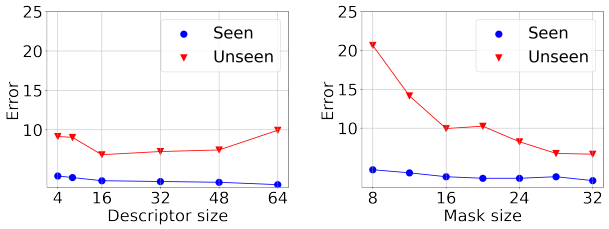

Figure 6: **Influence of the local feature dimension C and of the resolution of the local features and masks.** Using a good resolution is much more important than using high-dimensional local features as this allows discarding background more precisely when computing the similarity score.

Figure 7: The "Cat" object is often barely visible in the test images of Occluded-LINEMOD, resulting in large errors.

4.4. Failure Cases

When evaluated on O-LM, both our method and [38, 2] fail on the "Cat" object. As shown in Figure 7, this object is small and particularly heavily occluded in this dataset.

5. Conclusion

We have presented an efficient approach to 3D object recognition and pose estimation that can generalize to new objects without the need for retraining and that is robust to occlusions. Our analysis has shown that a global representation, which discards the grid structure of images, is not robust to clutter and results in inaccurate pose predictions. Our method, based on local representations, has much better properties and can be made robust to occlusions. We hope that our analysis and our new approach will guide the development of more practical systems.

Acknowledgments. We thank Michaël Ramamonjisoa, Tom Monnier, Elliot Vincent and Romain Loiseau for valuable feedback. This research was produced within the framework of Energy4Climate Interdisciplinary Center (E4C) of IP Paris and Ecole des Ponts ParisTech. This research was supported by 3rd Programme d'Investissements d'Avenir [ANR-18-EUR-0006-02]. This action benefited from the support of the Chair "Challenging Technology for Responsible Energy" led by l'X – Ecole polytechnique and the Fondation de l'Ecole polytechnique, sponsored by TOTAL. This work has received funding from the CHISTERA IPALM project and was performed using HPC resources from GENCI–IDRIS 2021-AD011012294R1.

References

[1] Seungryul Baek, Kwang In Kim, and Tae-Kyun Kim. Weakly-Supervised Domain Adaptation via GAN and Mesh Model for Estimating 3D Hand Poses Interacting Objects. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 1

[2] Vassileios Balntas, Andreas Doumanoglou, Caner Sahin, Juil Sock, Rigas Kouskouridas, and Tae-Kyun Kim. Pose Guided RGBD Feature Learning for 3D Object Pose Estimation. In *International Conference on Computer Vision (ICCV)*, 2017. 1, 2, 3, 4, 5, 6, 7, 8

[3] Eric Brachmann, Alexander Krull, Frank Michel, Stefan Gumhold, Jamie Shotton, and Carsten Rother. Learning 6D Object Pose Estimation Using 3D Object Coordinates. In *European Conference on Computer Vision (ECCV)*, 2014. 2, 5

[4] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised Learning of Visual Features by Contrasting Cluster Assignments. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020. 5

[5] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A Simple Framework for Contrastive Learning of Visual Representations. In *International Conference on Machine Learning (ICML)*, 2020. 3, 5

[6] Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey Hinton. Big Self-Supervised Models are Strong Semi-Supervised Learners. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020. 5

[7] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved Baselines with Momentum Contrastive Learning. *ArXiv*, 2020. 5

[8] Maximilian Denninger, Martin Sundermeyer, Dominik Winkelbauer, Youssef Zidan, Dmitry Olefir, Mohamad Elbadrawy, Ahsan Lodhi, and Harinandan Katam. Blenderproc. *arXiv preprint arXiv:1911.01911*, 2019. 5

[9] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross B. Girshick. Momentum Contrast for Unsupervised Visual Representation Learning. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 3, 5, 6

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 7

[11] Stefan Hinterstoisser, Vincent Lepetit, Slobodan Ilic, Stefan Holzer, G. Bradski, Kurt Konolige, and Nassir Navab. Model Based Training, Detection and Pose Estimation of Texture-Less 3D Objects in Heavily Cluttered Scenes. In *Asian Conference on Computer Vision (ACCV)*, 2012. 2, 5, 6

[12] R Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Phil Bachman, Adam Trischler, and Yoshua Bengio. Learning Deep Representations by Mutual Information Estimation and Maximization. In *International Conference on Learning Representations (ICLR)*, 2019. 3

[13] Tomas Hodan, Pavel Haluza, Stepan Obdrzalek, Jiri Matas, Manolis Lourakis, and Xenophon Zabulis. T-LESS: An RGB-D Dataset for 6D Pose Estimation of Texture-Less Objects. In *Winter Conference on Applications of Computer Vision (WACV)*, 2017. 2, 5, 6

[14] Tomas Hodan, Martin Sundermeyer, Bertram Drost, Yann Labbé, Eric Brachmann, Frank Michel, Carsten Rother, and Jiri Matas. BOP Challenge 2020 on 6D Object Localization. In *European Conference on Computer Vision (ECCV)*, 2020. 2

[15] Yinlin Hu, Pascal Fua, Wei Wang, and Mathieu Salzmann. Single-Stage 6D Object Pose Estimation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2

[16] Yinlin Hu, Joachim Hugonot, Pascal Fua, and Mathieu Salzmann. Segmentation-Driven 6D Object Pose Estimation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 1, 2

[17] Wadim Kehl, Fabian Manhardt, Federico Tombari, Slobodan Ilic, and Nassir Navab. SSD-6D: Making RGB-Based 3D Detection and 6D Pose Estimation Great Again. In *International Conference on Computer Vision (ICCV)*, 2017. 1, 2, 6

[18] Yann Labbé, Justin Carpentier, Mathieu Aubry, and Josef Sivic. CosyPose: Consistent Multi-View Multi-Object 6D Pose Estimation. In *European Conference on Computer Vision (ECCV)*, 2020. 1, 2

[19] Yi Li, Gu Wang, Xiangyang Ji, Yu Xiang, and Dieter Fox. DeepIM: Deep Iterative Matching for 6D Pose Estimation. In *European Conference on Computer Vision (ECCV)*, 2018. 1, 2

[20] Zhigang Li, Gu Wang, and Xiangyang Ji. CDPN: Coordinates-Based Disentangled Pose Network for Real-Time RGB-Based 6-DoF Object Pose Estimation. In *International Conference on Computer Vision (ICCV)*, 2019. 2

[21] David Lowe. Three-Dimensional Object Recognition from Single Two-Dimensional Images. *AI*, 31(3):355–395, 1987. 2

[22] Ishan Misra and Laurens Van Der Maaten. Self-Supervised Learning of Pretext-Invariant Representations. *CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 5

[23] Markus Oberweger, Mahdi Rad, and Vincent Lepetit. Making Deep Heatmaps Robust to Partial Occlusions for 3D Object Pose Estimation. In *European Conference on Computer Vision (ECCV)*, 2018. 1, 2

[24] Aaron Oord, Yazhe Li, and Oriol Vinyals. Representation Learning with Contrastive Predictive Coding. *ArXiv*, 2018. 3, 4, 5

[25] Kiru Park, Timothy Patten, and Markus Vincze. Pix2pose: Pixel-Wise Coordinate Regression of Objects for 6D Pose Estimation. In *International Conference on Computer Vision (ICCV)*, 2019. 2

[26] Kiru Park, Timothy Patten, and Markus Vincze. Neural Object Learning for 6D Pose Estimation Using a Few Cluttered Images. In *European Conference on Computer Vision (ECCV)*, 2020. 2

[27] Sida Peng, Yuan Liu, Qixing Huang, Xiaowei Zhou, and Hujun Bao. PVNet: Pixel-Wise Voting Network for 6DoF Pose Estimation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 1, 2

[28] Giorgia Pitteri, Aurélie Bugeau, Slobodan Ilic, and Vincent Lepetit. 3D Object Detection and Pose Estimation of Unseen Objects in Color Images with Local Surface Embeddings. In *Asian Conference on Computer Vision (ACCV)*, 2020. 1, 2

[29] Mahdi Rad and Vincent Lepetit. BB8: A Scalable, Accurate, Robust to Partial Occlusion Method for Predicting the 3D Poses of Challenging Objects Without Using Depth. In *International Conference on Computer Vision (ICCV)*, 2017. 1, 2

[30] Martin Sundermeyer, Maximilian Durner, En Yen Puang, Zoltan-Csaba Marton, Narunas Vaskevicius, Kai O. Arras, and Rudolph Triebel. Multi-Path Learning for Object Pose Estimation Across Domains. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 1, 2, 5, 6, 7

[31] Martin Sundermeyer, Zoltan-Csaba Marton, Maximilian Durner, Manuel Brucker, and Rudolph Triebel. Implicit 3D Orientation Learning for 6D Object Detection from RGB Images. In *European Conference on Computer Vision (ECCV)*, pages 699–715, 2018. 2, 5, 6, 7

[32] Martin Sundermeyer, Zoltan-Csaba Marton, Maximilian Durner, and Rudolph Triebel. Augmented Autoencoders: Implicit 3D Orientation Learning for 6D Object Detection. *IJCV*, 128(3):714–729, 2020. 1

[33] Bugra Tekin, Sudipta N. Sinha, and Pascal Fua. Real-Time Seamless Single Shot 6D Object Pose Prediction. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 1, 2

[34] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive Multiview Coding. In *European Conference on Computer Vision (ECCV)*, 2020. 3, 5

[35] Jonathan Tremblay, Thang To, Balakumar Sundaralingam, Yu Xiang, Dieter Fox, and Stan Birchfield. Deep Object Pose Estimation for Semantic Robotic Grasping of Household Objects. In *Conference on Robot Learning (CoRL)*, 2018. 1

[36] Laurens Van der Maaten and Geoffrey Hinton. Visualizing Data using t-SNE. *Journal of machine learning research*, 9(11), 2008. 3

[37] Wang, He and Sridhar, Srinath and Huang, Jingwei and Valentin, Julien and Song, Shuran and Guibas, Leonidas J. Normalized Object Coordinate Space for Category-Level 6D Object Pose and Size Estimation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 1, 2

[38] Paul Wohlhart and Vincent Lepetit. Learning Descriptors for Object Recognition and 3D Pose Estimation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 1, 2, 3, 4, 5, 6, 7, 8

[39] Zhirong Wu, Yuanjun Xiong, S. Yu, and D. Lin. Unsupervised Feature Learning via Non-parametric Instance Discrimination. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 3

[40] Jianxiong Xiao, James Hays, Krista A Ehinger, Aude Oliva, and Antonio Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *2010 IEEE computer society conference on computer vision and pattern recognition*, pages 3485–3492. IEEE, 2010. 6

[41] Yang Xiao, Yuming Du, and Renaud Marlet. PoseContrast: Class-Agnostic Object Viewpoint Estimation in the Wild with Pose-Aware Contrastive Learning. In *International Conference on 3D Vision (3DV)*, 2021. 3

[42] Yang Xiao, Xuchong Qiu, Pierre-Alain Langlois, Mathieu Aubry, and Renaud Marlet. Pose from Shape: Deep Pose Estimation for Arbitrary 3D Objects. In *British Machine Vision Conference*, 2019. 3

[43] Sergey Zakharov, Ivan S. Shugurov, and Slobodan Ilic. DPOD: 6D Pose Object Detector and Refiner. In *International Conference on Computer Vision (ICCV)*, 2019. 1, 2

[44] Xingyi Zhou, Arjun Karpur, Linjie Luo, and Qixing Huang. StarMap for Category-Agnostic Keypoint and Viewpoint Estimation. In *European Conference on Computer Vision (ECCV)*, 2018. 2