# Dual Pose-invariant Embeddings: Learning Category and Object-specific Discriminative Representations for Recognition and Retrieval

Rohan Sarkar, Avinash Kak
Electrical and Computer Engineering, Purdue University, USA
{sarkarr, kak}@purdue.edu

## Abstract

*In the context of pose-invariant object recognition and retrieval, we demonstrate that it is possible to achieve significant improvements in performance if both the category-based and the object-identity-based embeddings are learned simultaneously during training. In hindsight, that sounds intuitive because learning about the categories is more fundamental than learning about the individual objects that correspond to those categories. However, to the best of what we know, no prior work in pose-invariant learning has demonstrated this effect. This paper presents an attention-based dual-encoder architecture with specially designed loss functions that optimize the inter- and intra-class distances simultaneously in two different embedding spaces, one for the category embeddings and the other for the object level embeddings. The loss functions we have proposed are pose-invariant ranking losses that are designed to minimize the intra-class distances and maximize the inter-class distances in the dual representation spaces. We demonstrate the power of our approach with three challenging multi-view datasets, ModelNet-40, ObjectPI, and FG3D. With our dual approach, for single-view object recognition, we outperform the previous best by 20.0% on ModelNet40, 2.0% on ObjectPI, and 46.5% on FG3D. On the other hand, for single-view object retrieval, we outperform the previous best by 33.7% on ModelNet40, 18.8% on ObjectPI, and 56.9% on FG3D.*

## 1. Introduction

Pose-invariant recognition and retrieval [7] is an important problem in computer vision with practical applications in robotic automation, automatic checkout systems, and inventory management. The appearance of many objects belonging to the same general category can vary significantly from different viewpoints, and, yet, humans have no difficulty in recognizing them from arbitrary viewpoints. In pose-invariant recognition and retrieval, the focus is on mapping the object images to embedding vectors such that the embeddings for the objects that belong to the same category are pulled together for all the available viewpoints in relation to the embeddings for the objects for the different categories.

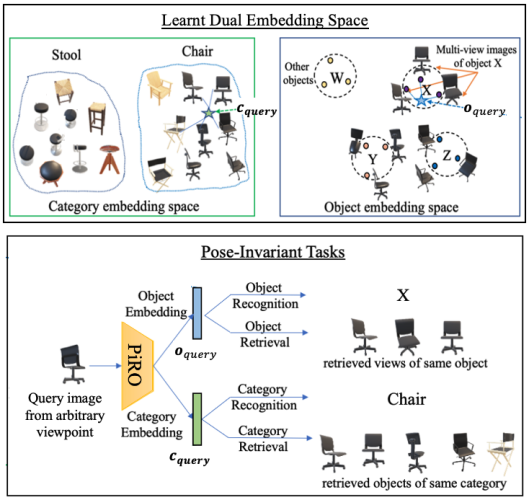



**Figure 1.** *The upper panel shows objects belonging to two different categories, chair and stool. In the proposed disentangled dual-space learning, the goal for the learning of category-based embeddings is to capture what maximally discriminates the objects belonging to the two categories — the presence or the absence of the back-rest. On the other hand, the object-identity based embeddings are meant to capture what is distinctive about each object. The lower panel illustrates our dual-space approach for simultaneously learning the embeddings in two different spaces for category and object-identity-based recognition and retrieval tasks.*

Our work demonstrates that the performance of pose-invariant learning as described above can be significantly improved if we disentangle the category-based learning from the object-identity-based learning.

Fig. 1 illustrates what we mean by disentangling the category-based representation from the object-identity-based representation. Assume that an object database contains images of different types of chairs and different types of stools. We would want our network to learn the category-based embedding vectors for the chair class and for the stool

class. These embeddings need to capture what is maximally discriminating between the chairs and the stools — the presence or the absence of a back-rest. At the same time, we would want the network to learn object-identity based embeddings. These embeddings should represent what is distinctive about each object type in relation to all other object types in the same category. For example, in the chair category, we would want the network to be able to discriminate between, say, lounge chairs and desk chairs.

Prior work [6, 10, 22, 26] has employed multi-view deep networks to learn aggregated multi-view representations capturing the variability in object appearance under different pose transformations. While these methods demonstrate good performance in category-level tasks when multiple views of objects are available during inference, *their performance degrades when only a single view is available*. Since real-world applications often necessitate inference from single views, Ho et al. [7] proposed a family of pose-invariant embeddings for both recognition and retrieval by imposing constraints such that the single-view embeddings of an object are clustered around its multi-view embeddings, which in turn are clustered around a proxy embedding representing the associated high-level category that the object belongs to. However, this approach does not do a good enough job of separating the embeddings for two different objects that belong to the same category (e.g., two different types of chairs, two different types of kettles, etc.), As a result, prior approaches perform well on category-level tasks but not on object-level tasks, as we will demonstrate later in our experimental results (see Tables 2, 3).

Here is arguably the most significant difference between the previous methods and the one being proposed in this paper: *Rather than learning representations that capture both category-specific and object-specific discriminative features within the same embedding space, we simultaneously learn them in two distinct embedding spaces*, as depicted in the lower panel in Fig. 1. In one space that is devoted to category-based representations, objects from the same category can be closely embedded together, capturing shared characteristics among them, while in the other space, the one for object identity-based representations, embeddings for the different object types (within the same category or otherwise) are allowed to be as separated as dictated by the attributes that differentiate them. This strategy enables our network to learn object representations that are more discriminative overall. This should explain the superior performance of our framework in both recognition and retrieval, especially for the more difficult case when only a single-viewpoint query image is available. For single-view object recognition, we get an improvement in accuracy of 20.0% on ModelNet40, 46.5% on FG3D, and 2.0% on ObjectPI. Along the same lines, for the case of single-view object retrieval, we achieve a significant mAP improvement of 33.7% on ModelNet-40, 56.9% on FG3D, and 18.8% on ObjectPI datasets.

In order to learn the dual embeddings simultaneously, we propose an encoder that we refer to as the Pose-invariant Attention Network (PAN). PAN uses a shared CNN backbone for capturing visual features common to both the category and the object-identity based representations from a set of images of an object recorded from different viewpoints. The visual features are then mapped to separate low-dimensional category and object-identity based embeddings using two fully connected layers. PAN also aggregates visual features of objects from different views using self-attention to generate what we call multi-view embeddings. The dual embeddings, defined in Section 3, can be used for *both category and object-level recognition and retrieval from single and multiple views.*

For training the network, we propose two pose-invariant category and object-identity based losses that are jointly optimized to learn the dual embeddings. The pose-invariant category loss clusters together the instances of different objects belonging to the same category while separating apart the instances from different categories in the category embedding space. On the other hand, the pose-invariant object-identity based loss clusters together the instances that carry the same object-identity label and separates what would otherwise be mutually confusing object instances with two different object-identity labels from the same category in the object embedding space.

## 2. Background and Related work

**(A) Ranking and Proxy-based Losses:** Ranking losses, used in deep metric learning, focus on optimizing the relative pair-wise distances between exemplars (pairs [2], triplets [8] or quadruplets [1]), such that similar samples are pulled closer and dissimilar samples are pushed apart. For ranking losses, the selection of informative exemplars [4, 5, 9, 16, 19, 21, 23, 24, 27] is crucial, which however incurs additional computational costs and memory. To reduce the training complexity, proxy-based approaches [14] define a proxy embedding for each class and optimize sample-to-proxy distances. However, they only capture relationships between samples and the proxies, which are less informative compared to the extensive sample-to-sample relations inherent in pair-based losses, which is particularly important for fine-grained tasks.

**(B) Multi-view and Pose-Invariant Classification and Retrieval:** In multi-view object recognition and retrieval [6, 10, 22, 26], each object from category $c$ is captured from a set of $V$ views and is denoted by $\mathbf{X} = \{\mathbf{x}_k\}_{k=1}^{V}$. For each object, a set of single-view embeddings are extracted by inputing each image $\mathbf{x}_k$ to a network $g_s$, which are then aggregated to generate multi-view embeddings as $g_m(\mathbf{X}) = \Phi(\{g_s(\mathbf{x}_k)\}_{k=1}^{V})$, where $\Phi$ denotes the aggre-

gation operation. Multi-view losses cluster the multi-view embeddings of objects from the same category together and yield good performance on category-based tasks when multiple views of objects are available during inference. However, they perform poorly when only a single view is available during inference as the single-view embeddings are not constrained to be close to the multi-view embeddings in the embedding space. To mitigate this, the approach by [7] learns pose-invariant embeddings by combining two separate view-to-object and object-to-category models trained using different types of pose-invariant losses. These losses optimize the pose-invariance distance defined as

$$d^{pi}(\mathbf{x}, \mathbf{X}, \mathbf{p}_c) = \alpha d(g_s(\mathbf{x}), g_m(\mathbf{X})) + \beta d(g_m(\mathbf{X}), \mathbf{p}_c) \quad (1)$$

where, $\alpha$ promotes the clustering of single-view embeddings around the object's multi-view embedding, while $\beta$ encourages the clustering of the multi-view embedding of the object around the learned proxy embedding $\mathbf{p}_c$ for its category $c$. However, these losses do not effectively separate embeddings of distinct objects from the same category, as we will demonstrate later in Fig. 5. This results in poor performance on object-based tasks.

In summary, prior work focused primarily on learning category-specific embeddings, with the object-to-object variations within each category represented by the variations in the embedding vectors within the same embedding space. In contrast, we learn a unified model that explicitly decouples the object and category embeddings. The model is trained jointly using two proposed pose-invariant ranking losses. In the category embedding space, the proposed loss clusters instances of different objects belonging to the same category together. In the object embedding space, the proposed loss clusters different views of the same object while separating confusing instances of different objects from the same category, thereby capturing discriminatory features to distinguish between similar objects from the same category. This significantly improves object recognition and retrieval performance over prior methods (ref. Tables 2, 3).

**(C) Attention-based Architectures:** Since the advent of ViT [3], transformers have become increasingly popular for a variety of computer vision tasks. Most relevant to our work are hybrid architectures comprising a CNN backbone in conjunction with a transformer encoder that use multi-head attention layers to learn aggregated representations from image collections comprising different items [17, 18] and multi-view 3D shape representations [15, 25] for classification and retrieval tasks. In contrast, we only use a single-head self-attention layer for each subspace to aggregate visual features extracted from a DNN across different views to learn multi-view embeddings. The architectures in [15, 25] learn multi-view shape representations for category-based tasks and require multi-view images at inference time. In contrast, our dual-encoder is designed to simultaneously learn pose-invariant category and object representations that can be utilized for both category and object-based tasks from single and multiple views during inference. Separate models for classification and retrieval tasks were proposed in [18], whereas our unified model can address both tasks jointly. Also, positional encodings are utilized by [3, 15] to preserve input order, but not by [18]. We omit positional encodings to ensure that the learned representations are independent of the input view order.

## 3. Proposed Approach

A high-level overview of our framework PiRO for learning dual pose-invariant representations of objects is shown in Fig. 2. Our approach learns by comparing pairs of objects belonging to the same category, while taking into account their multi-view appearances. This is illustrated in Fig. 2 where we show two different kettles, obviously belonging to the same category, and, in the depiction in the figure, we use three randomly chosen viewpoint images for each kettle. For the purpose of explanation, we have labeled the two objects as $a$ and $b$. In general, we choose $V$ number of randomly selected images from the different viewpoints for each object. The objective of this within-category learning is to become aware of the common attributes shared by these objects, like the spout, lid, handle, and overall body structure, enabling their categorization as a kettle.

The multi-view images are input to our proposed dual-encoder PAN, which we introduce in Section 3.1. The dual encoder consists of a shared CNN backbone responsible for capturing common visual features, along with two distinct heads dedicated to the learning of the dual category and object-identity based embeddings.

The encoder is trained jointly using pose-invariant losses designed for each respective embedding space, as described in Section 3.2. In the category embedding space, the loss is designed to cluster together the embeddings of the ob-

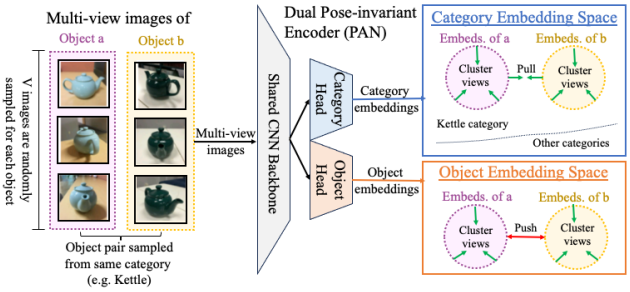



**Figure 2.** *An overview of our PiRO framework to learn the dual pose-invariant object and category embeddings using losses specifically designed for each embedding space. Multi-view images of two randomly chosen objects from the same category are used to learn common characteristics of the objects in the category embedding space and discriminatory attributes to distinguish between them in the object embedding space.*

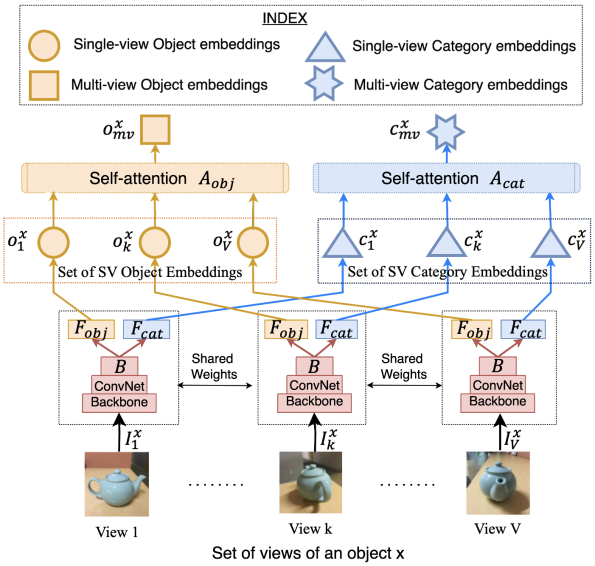



**Figure 3.** *The Pose-invariant Attention Network (PAN) takes a set of multi-view images of an object as input, producing both single-view and multi-view embeddings for each representational subspace. The object embeddings are depicted in orange, while the category embeddings are in blue.*

jects from the same category regardless of the viewpoints, as shown in top-right of Fig. 2. On the other hand, in the object-identity embedding space, the loss is designed to cluster together the embeddings for the instances that carry the same object-identity label, again regardless of the viewpoints, while separating instances with different object-identity labels from the same category, as shown in bottom-right of Fig. 2. The idea is for the encoder to capture shared characteristics among objects within the same category in the category space and discriminatory attributes to distinguish between them in the object space. These dual embeddings can then be utilized for pose-invariant category and object-based recognition and retrieval.

### 3.1. Pose-invariant Attention Network (PAN):

Fig. 3 illustrates in greater detail the design of PAN, the Pose-invariant Encoder shown previously in Fig. 2. It consists of a CNN backbone ($B$), two FC layers ($F_{obj}$ and $F_{cat}$) and two single-head self-attention layers ($A_{obj}$ and $A_{cat}$). It takes as input an unordered set of images from $V$ different views of an object $x$ from category $l_x$ represented as $\mathbf{I}^x_{set} = \{\mathbf{I}^x_1, \cdots, \mathbf{I}^x_k, \cdots, \mathbf{I}^x_V\}$. The backbone and FC layers for each view share the same weights.

The backbone learns visual features common to both the category and object-identity representations. The visual features extracted from each object view are subsequently input to the FC layer ($F_{obj}$) to generate the object-identity embeddings. The set of *single-view object embeddings* for object $x$ is denoted by:

$$\mathcal{E}^x_{obj} = \{\mathbf{o}^x_k \mid \mathbf{o}^x_k = F_{obj}(B(\mathbf{I}^x_k)) \quad \forall \mathbf{I}^x_k \in \mathbf{I}^x_{set}\} \qquad (2)$$

Similarly, the shared visual features are input to another FC layer ($F_{cat}$) to generate category embeddings. The set of *single-view category embeddings* for object $x$ is denoted by:

$$\mathcal{E}^x_{cat} = \{\mathbf{c}^x_k \mid \mathbf{c}^x_k = F_{cat}(B(\mathbf{I}^x_k)) \quad \forall \mathbf{I}^x_k \in \mathbf{I}^x_{set}\} \qquad (3)$$

The single-view object and category embeddings are then passed into the self-attention layers $A_{obj}$ and $A_{cat}$ to learn the corresponding multi-view embeddings. The self-attention mechanism allows weighted interactions between the features extracted from one view with features extracted from all the remaining views in the set to capture the correlation between visual features across multiple views effectively. The resulting feature vectors from the images of an object are then aggregated using mean-pooling to get the multi-view embeddings. The resulting *multi-view object and category embedding* for object $x$ is denoted by:

$$\mathbf{o}^x_{mv} = \frac{1}{V}\sum_{k=1}^{V} A_{obj}(\mathcal{E}^x_{obj}), \mathbf{c}^x_{mv} = \frac{1}{V}\sum_{k=1}^{V} A_{cat}(\mathcal{E}^x_{cat}) \qquad (4)$$

### 3.2. Pose-invariant Losses

The single-view and multi-view embeddings extracted using PAN are used for constructing pose-invariant losses that train the encoder to map object images across different viewpoints to compact low-dimensional subspaces, where the Euclidean distance between embeddings corresponds to a measure of object similarity across viewpoints. We propose two such pose-invariant losses for the object and category embedding spaces next.

**(A) Pose-invariant Object Loss:** This loss is designed specifically for fine-grained object recognition and retrieval from arbitrary viewpoints. The loss pulls together the embeddings of the different views of the same object, as shown by the green arrows in Fig. 4(A). This allows the encoder to learn common view-invariant features from multiple views. At the same time, it is designed to increase the inter-class distances between the embeddings (as shown by the red arrows in the same figure). That allows the encoder to learn the discriminative features to distinguish between visually similar objects from the same category.

Let us consider a pair of objects $(a, b)$ from the same high-level category as shown in Fig. 4(A). The object-identity embeddings generated by the encoder (ref. Eqn. 2) from $V$ views for each of the objects are symbolically represented as the two sets $\mathcal{E}^a_{obj}$ and $\mathcal{E}^b_{obj}$ respectively. For each such pair, embeddings of different objects with the minimum separation between them are the most informative and are chosen as the *confusers*. These embeddings are called confusers because they maximally violate the inter-class margin between the object pair and are the most likely to confuse a classifier. The confusers denoted by $\mathbf{o}^a_{con}$ and

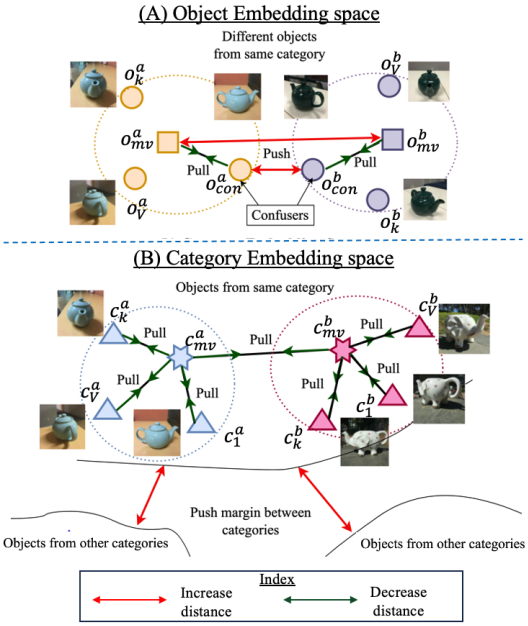



**Figure 4.** *The pose-invariant losses enhance intra-class compactness and inter-class separation in the dual embedding spaces. In the object embedding space (top), confusing instances of two different objects from the same category are separated. In the category embedding space (bottom), objects belonging to the same category are pulled closer while being separated from those belonging to other categories.*

$\mathbf{o}^b_{con}$ are computed as

$$\mathbf{o}^a_{con}, \mathbf{o}^b_{con} = \underset{\forall \mathbf{x} \in \mathcal{E}^a_{obj}, \forall \mathbf{y} \in \mathcal{E}^b_{obj}}{\operatorname{argmin}} d(\mathbf{x}, \mathbf{y}) \tag{5}$$

where, $d(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|_2$ is the euclidean distance between the embeddings $\mathbf{x}$ and $\mathbf{y}$. The multi-view object embeddings $\mathbf{o}^a_{mv}, \mathbf{o}^b_{mv}$ from the respective object-identity classes are considered as *positives*.

The intra-class compactness and inter-class separability are controlled using two margins $\alpha$ and $\beta$ respectively. Our pose-invariant object-identity loss has two components:

**(i) Clustering loss** ensures that the distance between the multi-view embedding and the single-view confuser embedding in Eqn. 5 of the same object-identity class $a$ does not exceed the margin $\alpha$. For the object-identity class $a$, it is defined as:

$$\mathcal{L}^a_{intra} = \Big[ d(\mathbf{o}^a_{mv}, \mathbf{o}^a_{con}) - \alpha \Big]_+ \tag{6}$$

where, $[z]_+ = max(z, 0)$ is the hinge loss.

**(ii) Separation loss** ensures that the minimum distance between the single-view confuser embeddings of two objects $a$ and $b$ and also the separation between the multi-view object embeddings of the corresponding objects is greater than a margin $\beta$. By separating the confusers and multi-view embeddings of two objects from the same category, the encoder will learn discriminatory features. It is defined as:

$$\mathcal{L}^{a,b}_{inter} = \Big[ \beta - d(\mathbf{o}^a_{con}, \mathbf{o}^b_{con}) \Big]_+ + \Big[ \beta - d(\mathbf{o}^a_{mv}, \mathbf{o}^b_{mv}) \Big]_+ \tag{7}$$

The overall loss is defined as:

$$\mathcal{L}^{a,b}_{piobj} = \mathcal{L}^a_{intra} + \mathcal{L}^b_{intra} + \mathcal{L}^{a,b}_{inter} \tag{8}$$

**(B) Pose-invariant Category Loss:** As shown by the green arrows in Fig. 4(B), this loss ensures that in the category embedding space, the single-view and multi-view embeddings of an object are well clustered and the multi-view embeddings for two different object-identity classes from the same category are embedded close to each other and do not exceed a margin $\theta$. The clustering loss for the category embeddings (ref. Eqn. 3) of the objects $a, b$ from the same category is defined as:

$$\mathcal{L}^{a,b}_{picat} = \Big[ d^a_{sm} - \theta \Big]_+ + \Big[ d^b_{sm} - \theta \Big]_+ + \Big[ d(\mathbf{c}^a_{mv}, \mathbf{c}^b_{mv}) - \theta \Big]_+ \tag{9}$$

where, $d^x_{sm} = \frac{1}{V} \sum_{k=1}^{V} d(\mathbf{c}^x_k, \mathbf{c}^x_{mv})$ is the mean of the distances between the multi-view and single-view embeddings for an object $x$ in the category embedding space.

**(C) Total Loss:** In the category embedding space, we use the large-margin softmax (L-Softmax) loss for separating the embeddings of objects from different categories (shown by the red arrows in Fig. 4(B)) using a margin $\gamma$. The dual-encoder PAN is jointly trained using all the losses and the overall loss is defined as

$$\mathcal{L} = \frac{1}{|\mathcal{P}|} \sum_{(a,b) \in \mathcal{P}} \mathcal{L}^a_{cat} + \mathcal{L}^b_{cat} + \mathcal{L}^{a,b}_{picat} + \mathcal{L}^{a,b}_{piobj} \tag{10}$$

where, $\mathcal{L}^x_{cat} = \frac{1}{V} \sum_{k=1}^{V} \mathcal{L}_\gamma(\mathbf{c}^x_k, l_x)$ such that $L_\gamma(\mathbf{c}^x_k, l_x)$ is the L-Softmax loss [11] with margin $\gamma$ for a category embedding $\mathbf{c}^x_k$ of an object $x$ belonging to category $l_x$ from any viewpoint $k$, and $\mathcal{P}$ is the set of all object pairs where each pair is randomly sampled from the same category.

## 4. Experiments

In this section, we evaluate our approach on pose-invariant classification and retrieval (PICR) tasks on three multi-view object datasets, report ablation studies at the end of this section, and additional results in the supplementary material.

**(A) Setup, Implementation Details, and Results:**

**Datasets:** ModelNet-40 [26] is a multi-view dataset comprising 3983 objects (3183 train and 800 test) with roughly 100 unique CAD models per category from 40 common object categories with 12 views per model, generated by starting from an arbitrary pose and rotating each model every 30 degrees. The ObjectPI dataset [7] consists of images collected in the wild, by placing each object in a natural scene

| Dataset | Embed. Space | Classification (Accuracy %) | | | | | Retrieval (mAP %) | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Category | | Object | | **Average** | Category | | Object | | **Average** |
| | | Single-view | Multi-view | Single-view | Multi-view | | Single-view | Multi-view | Single-view | Multi-view | |
| ObjectPI | Single | $69.56 \pm 0.9$ | $80.27 \pm 1.9$ | $88.35 \pm 0.3$ | $98.98 \pm 0.9$ | $84.29 \pm 0.7$ | $65.81 \pm 0.5$ | $75.60 \pm 0.7$ | $68.55 \pm 0.5$ | $99.46 \pm 0.5$ | $77.35 \pm 0.4$ |
| | Dual | $70.22 \pm 0.7$ | $82.48 \pm 1.0$ | $93.07 \pm 0.8$ | $98.64 \pm 0.5$ | $\mathbf{86.10 \pm 0.3}$ | $65.20 \pm 0.4$ | $82.80 \pm 0.5$ | $80.61 \pm 0.5$ | $99.46 \pm 0.3$ | $\mathbf{82.02 \pm 0.3}$ |
| ModelNet | Single | $85.09 \pm 0.3$ | $88.08 \pm 0.6$ | $82.90 \pm 1.5$ | $86.75 \pm 1.2$ | $85.71 \pm 0.5$ | $78.88 \pm 0.2$ | $82.88 \pm 0.2$ | $61.89 \pm 2.3$ | $91.22 \pm 0.8$ | $78.71 \pm 0.7$ |
| | Dual | $84.96 \pm 0.2$ | $88.32 \pm 0.4$ | $94.14 \pm 0.3$ | $96.88 \pm 0.2$ | $\mathbf{91.07 \pm 0.2}$ | $79.30 \pm 0.2$ | $85.28 \pm 0.4$ | $84.46 \pm 0.2$ | $98.17 \pm 0.1$ | $\mathbf{86.80 \pm 0.1}$ |
| FG3D | Single | $78.18 \pm 0.2$ | $80.42 \pm 0.1$ | $26.51 \pm 0.3$ | $29.76 \pm 0.7$ | $53.72 \pm 0.3$ | $65.05 \pm 0.3$ | $69.28 \pm 0.2$ | $15.79 \pm 0.1$ | $41.98 \pm 0.6$ | $48.02 \pm 0.3$ |
| | Dual | $78.89 \pm 0.2$ | $81.81 \pm 0.1$ | $83.00 \pm 0.2$ | $91.56 \pm 0.1$ | $\mathbf{83.81 \pm 0.1}$ | $67.95 \pm 0.3$ | $74.24 \pm 0.3$ | $72.78 \pm 0.3$ | $95.47 \pm 0.1$ | $\mathbf{77.61 \pm 0.2}$ |

**Table 1.** *Pose-invariant Classification and Retrieval results on category and object-level tasks using our method for single and dual embedding spaces on the ModelNet-40, FG3D and ObjectPI datasets. The average performance along with standard deviation are reported.*

and capturing pictures from 8 views around the object, for 480 objects (382 train and 98 test) from 25 categories. We use the same training and test splits provided by [7] for both datasets. Additionally, we also evaluate our method on FG3D [12] which is a large-scale dataset for fine-grained object recognition with 12 views per object for 25552 objects (21575 training and 3977 test) from 66 categories.

**Tasks:** Ho et al. [7] proposed five tasks: *Single-view and multi-view category recognition*. These tasks predict the category from a single view and a set of object views respectively. *Single-view and multi-view category retrieval*. The goal of these tasks is to retrieve images from the same category as the query object from a single view and multiple views respectively. *Single-view object retrieval*. This task aims to retrieve other views of the same object in the query view. We additionally report results using our method in Table 1 on three more tasks which are extensions of the above-mentioned tasks. These tasks are *single and multi-view object recognition* and *multi-view object retrieval*. The details of these tasks are provided in Supplemental Sec. 12. Classification and retrieval performance are reported as accuracy and mean average precision (mAP) respectively.

**Training Details:** Images are resized to $224 \times 224$ and normalized before being input to the network. The VGG-16 network [20] is used as the CNN backbone for a fair comparison with other state-of-the-art approaches. The last FC layers are modified to generate 2048-D embeddings and are initialized with random weights. A single layer and single head self-attention layer is used with a dropout of 0.25. The network is jointly trained using the proposed pose-invariant category and object losses. For all datasets, we set the margins $\alpha = 0.25, \beta = 1.00$ for the object embedding space and margins $\theta = 0.25, \gamma = 4.00$ for the category embedding space. We use the Adam optimizer with a learning rate of $1e^{-5}$ for ObjectPI, ModelNet-40, and $5e^{-5}$ for FG3D. We train for 25 epochs and use the step scheduler that reduces the learning rate by half after every 5 epochs. Our code is available at https://github.com/sarkar-rohan/PiRO.

**(B) Comparison with the State-of-the-art:** In Table 2, we compare performance of our method against several state-of-the-art multi-view and pose-invariant methods [6, 7, 22] reported by [7] on the ModelNet-40 and ObjectPI datasets. For the single-view object recognition task, we report the results using the trained models provided by [7]. As explained in Section 2(B), the multi-view methods are designed for category-based tasks when multiple images are available during inference. However, they perform poorly when only a single view is available and Pose-invariant (PI) methods outperform the multi-view (MV) methods on single-view tasks as they constrain the single-view embeddings to be clustered close to the multi-view embeddings. Although these pose-invariant methods encourage the clustering of different views of the same object, they don't effectively separate the confusing instances of neighboring objects from the same category in the embedding space. Hence, they don't capture discriminative features to distinguish between visually similar objects from the same category because of which they exhibit poor performance on the single-view object recognition and retrieval tasks.

We observe that our method PiRO-DE when learning dual category and object embeddings, outperforms the state-of-the-art methods on both the average classification (improvement of 7.7% on ModelNet-40 and 2.6% on ObjectPI) and retrieval tasks (improvement of 13.0% on ModelNet-40 and 8.8% on ObjectPI). We notice a significant improvement in the single-view object recognition (accuracy improves by 20.0% on ModelNet40 and 2.0% on ObjectPI) and retrieval tasks (mAP improves by 33.7% on ModelNet-40 and 18.8% on ObjectPI). Even in the single embedding space, PiRO-SE shows improvements on object-based tasks compared to the state-of-the-art approaches.

We train the state-of-the-art pose-invariant methods [7] on the FG3D dataset and compare performance with our method in Table 3. FG3D is more challenging for object-level tasks as it comprises a large number of similar objects in each category with fine-grained differences. As mentioned earlier, prior methods mainly focus on learning category-specific embeddings and do not effectively

| Method | ModelNet-40 (12 views) | | | | | | | | ObjectPI (8 views) | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Classification (Accuracy %) | | | | Retrieval (mAP %) | | | | Classification (Accuracy %) | | | | Retrieval (mAP %) | | | |
| | SV Cat | MV Cat | SV Obj | **Avg** | SV Cat | MV Cat | SV Obj | **Avg** | SV Cat | MV Cat | SV Obj | **Avg** | SV Cat | MV Cat | SV Obj | **Avg** |
| MV-CNN | 71.0 | 87.9 | 65.6 | 74.8 | 41.7 | 71.5 | 29.6 | 47.6 | 62.1 | 74.1 | 75.8 | 70.7 | 53.8 | 72.3 | 42.6 | 56.2 |
| PI-CNN | **85.4** | 88.0 | 65.1 | 79.5 | 77.5 | 81.8 | *50.8* | *70.0* | 66.5 | 76.5 | 61.6 | 68.2 | 58.9 | 72.1 | 60.7 | 63.9 |
| MV-TC | 77.3 | 88.9 | 54.2 | 73.5 | 63.5 | 84.0 | 36.6 | 61.4 | 65.7 | 79.2 | 65.9 | 70.3 | 59.5 | *77.3* | 51.8 | 62.9 |
| PI-TC | 81.2 | 88.9 | *74.1* | *81.4* | 71.5 | *84.2* | 41.4 | 65.7 | *69.3* | 77.5 | 91.1 | *79.3* | *63.8* | 76.7 | *61.8* | *67.4* |
| MV-Proxy | 79.7 | **89.6** | 37.1 | 68.8 | 66.1 | 85.1 | 35.0 | 62.1 | 63.2 | 78.3 | 53.6 | 65.0 | 57.9 | 74.7 | 49.3 | 60.6 |
| PI-Proxy | 85.1 | *88.7* | 66.1 | 80.0 | **79.9** | 85.1 | 40.6 | 68.5 | 68.7 | *80.0* | 70.8 | 73.2 | 62.6 | 78.2 | 49.4 | 63.4 |
| PiRO-SE (Ours) | 85.1 | 88.1 | 82.9 | 85.4 | *78.9* | 82.9 | 61.9 | 74.6 | 69.6 | 80.3 | *88.4* | 79.4 | **65.8** | 75.6 | 68.5 | 70.0 |
| PiRO-DE (Ours) | *85.0* | 88.3 | **94.1** | **89.1** | 79.3 | **85.3** | **84.5** | **83.0** | **70.2** | **82.5** | **93.1** | **81.9** | 65.2 | **82.8** | **80.6** | **76.2** |

**Table 2.** *Comparison of performance on pose-invariant classification and retrieval tasks on the ObjectPI and ModelNet-40 datasets with the state-of-the-art approaches. The best, second-best, and third-best performance is highlighted in bold, underline, and italics respectively. The methods starting with MV indicate multi-view methods and those starting with PI indicate methods that learn pose invariant embeddings. For our method PiRO, SE and DE stands for single and dual embedding spaces. The average classification and retrieval performance indicate that we learn better representations for recognition and retrieval tasks on both datasets. The improvements in single-view object recognition and retrieval performance are the most significant.*

| Method | Classification (Accuracy %) | | | | Retrieval (mAP %) | | | |
|---|---|---|---|---|---|---|---|---|
| | SV Cat | MV Cat | SV Obj | **Avg** | SV Cat | MV Cat | SV Obj | **Avg** |
| PI-CNN | 79.7 | **83.3** | 23.6 | 62.2 | 70.2 | 76.8 | 10.5 | 52.5 |
| PI-Proxy | **80.0** | 83.2 | 23.4 | 62.2 | **70.6** | **77.0** | 10.7 | 52.8 |
| PI-TC | 76.1 | 82.5 | 36.5 | 65.0 | 61.5 | 74.7 | 15.9 | 50.7 |
| Ours | 78.9 | 81.8 | **83.0** | **81.2** | 68.0 | 74.2 | **72.8** | **71.7** |

**Table 3.** *Comparison of performance on the FG3D dataset with state-of-the-art pose-invariant methods.*

separate the embeddings for objects within each category. In contrast, our proposed pose-invariant object loss separates confusing instances of objects from the same category which helps learn more discriminative fine-grained features to distinguish between visually similar objects resulting in significant improvement on single-view object recognition accuracy of 46.5% and object retrieval mAP of 56.9%. Overall, we outperform the pose-invariant methods on the classification tasks by 16.2% and retrieval tasks by 18.9%.

**(C) Ablation Studies:**

**Visualization of Pose-invariant Embeddings:** From Fig. 5, we observe that for the pose-invariant methods (PI-CNN, PI-Proxy, and PI-TC), the embeddings for objects from the same category are not well-separated leading to poor performance on object-based tasks. In contrast, the object embeddings generated using our method are much better separated as our pose-invariant object loss separates confusing instances of objects from the same category. A more detailed comparison is shown in Supplemental Sec. 6.

**Single and Dual Embedding Spaces:** From Tables 1 and 4, we observe that learning dual embeddings leads to better overall performance, especially for object-based tasks. This is because, for category-based tasks, we aim to embed objects from the same category close to each other while for object-based tasks, we aim to separate objects apart from each other to be able to discriminate between them. This leads to contradicting goals for object and category-based tasks in the single embedding space. Learning dual embeddings more effectively captures category and object-specific attributes in separate representation spaces leading to overall performance improvements.

**Pose-invariant Losses:** We employ three losses in PiRO: $\mathcal{L}_{cat}$ to distinguish between different categories, $\mathcal{L}_{picat}$ for clustering objects from the same category, and $\mathcal{L}_{piobj}$ for clustering different views of the same object and separating confusing instances from different objects of the same cat-

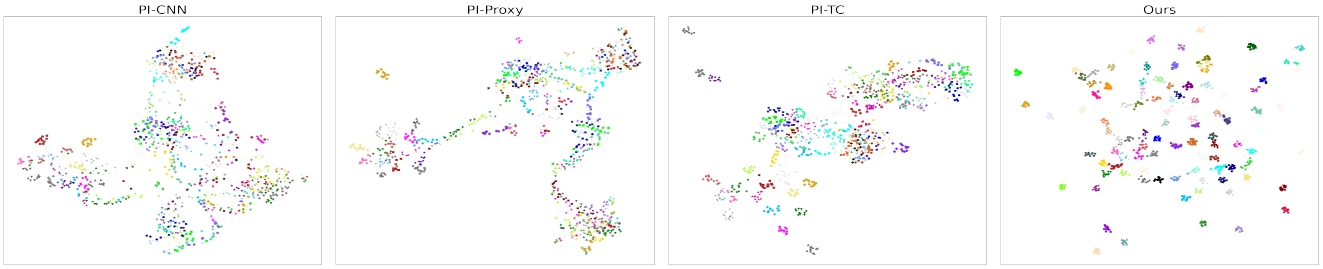



**Figure 5.** *We show UMAP [13] visualizations for a qualitative comparison of the object embedding space learned for the ModelNet40 test dataset (from 5 categories such as table, desk, chair, stool, and sofa with 100 objects) by prior pose-invariant methods [7] and our method. Each instance is an object view and a unique color and shape is used to denote each object-identity class in the visualizations.*

| Dataset | Embed. Space | Losses | Classification (Accuracy %) | | | | | Retrieval (mAP %) | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | Category | | Object | | Avg. | Category | | Object | | Avg. |
| | | | SV | MV | SV | MV | | SV | MV | SV | MV | |
| Object PI | Single | $\mathcal{L}_{cat}$ | 70.7 | 81.6 | 78.7 | 87.8 | 79.7 | 65.3 | 82.9 | 54.8 | 92.9 | 73.9 |
| | | $\mathcal{L}_{cat} + \mathcal{L}_{piobj}$ | 69.4 | 81.6 | 88.5 | 98.0 | 84.4 | **66.0** | 75.6 | 68.5 | 98.9 | 77.2 |
| | | $\mathcal{L}_{cat} + \mathcal{L}_{piobj} + \mathcal{L}_{picat}$ | 71.2 | 82.7 | 83.3 | 95.9 | 83.3 | 65.6 | 82.8 | 62.3 | 98.0 | 77.2 |
| | Dual | $\mathcal{L}_{cat} + \mathcal{L}_{piobj}$ | 71.2 | 82.7 | **94.5** | **99.0** | **86.8** | 65.7 | 82.9 | 80.5 | **99.5** | 82.2 |
| | | $\mathcal{L}_{cat} + \mathcal{L}_{piobj} + \mathcal{L}_{picat}$ | **71.3** | **83.7** | 92.7 | 98.0 | 86.4 | 65.7 | **83.4** | **81.0** | 99.0 | **82.3** |
| Model Net40 | Single | $\mathcal{L}_{cat}$ | 84.7 | 88.4 | 71.3 | 75.9 | 80.1 | 79.0 | 84.8 | 45.3 | 82.0 | 72.8 |
| | | $\mathcal{L}_{cat} + \mathcal{L}_{piobj}$ | **85.4** | 88.8 | 81.2 | 85.6 | 85.2 | 79.1 | 83.1 | 59.2 | 90.4 | 78.0 |
| | | $\mathcal{L}_{cat} + \mathcal{L}_{piobj} + \mathcal{L}_{picat}$ | 84.7 | 88.4 | 71.8 | 79.3 | 81.0 | 78.7 | 84.9 | 49.1 | 85.2 | 74.5 |
| | Dual | $\mathcal{L}_{cat} + \mathcal{L}_{piobj}$ | 84.5 | 88.6 | **94.6** | 96.6 | 91.1 | 78.9 | 85.0 | **85.2** | 98.1 | 86.8 |
| | | $\mathcal{L}_{cat} + \mathcal{L}_{piobj} + \mathcal{L}_{picat}$ | 85.2 | **88.9** | 93.7 | **96.9** | **91.2** | **79.7** | **86.1** | 84.0 | **98.2** | **87.0** |
| FG3D | Single | $\mathcal{L}_{cat}$ | **79.3** | 81.8 | 18.2 | 19.0 | 49.5 | 66.6 | 73.1 | 9.7 | 28.4 | 44.5 |
| | | $\mathcal{L}_{cat} + \mathcal{L}_{piobj}$ | 78.3 | 80.2 | 26.2 | 31.0 | 53.9 | 64.9 | 69.0 | 15.7 | 42.9 | 48.1 |
| | | $\mathcal{L}_{cat} + \mathcal{L}_{piobj} + \mathcal{L}_{picat}$ | 78.4 | 81.1 | 29.3 | 41.8 | 57.6 | 65.1 | 70.8 | 17.9 | 55.0 | 52.2 |
| | Dual | $\mathcal{L}_{cat} + \mathcal{L}_{piobj}$ | 78.7 | **82.2** | **83.2** | 91.4 | **83.9** | 67.6 | 73.1 | 72.8 | 95.3 | 77.2 |
| | | $\mathcal{L}_{cat} + \mathcal{L}_{piobj} + \mathcal{L}_{picat}$ | 79.0 | 81.9 | 83.1 | **91.6** | **83.9** | **68.1** | **74.4** | **73.0** | **95.5** | **77.8** |

**Table 4.** *Ablations of the proposed losses in the single and dual embedding spaces.*

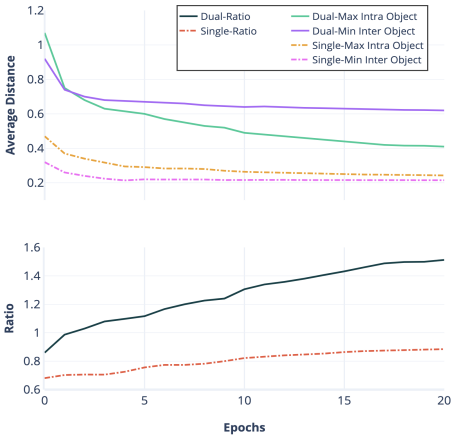


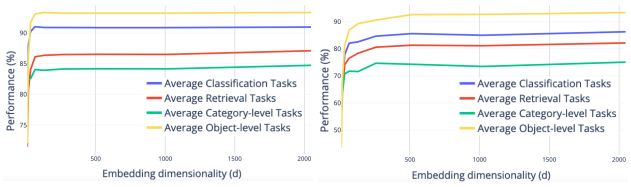

**Figure 6.** *Optimization of the inter-class and intra-class distances for object-identity classes during training while learning single and dual embedding spaces for the ModelNet40 dataset.*

egory for object-based tasks. Table 4 shows that in the single embedding space, $\mathcal{L}_{cat}$ is effective for category-based tasks, but not for object-based tasks. Adding $\mathcal{L}_{piobj}$ improves performance in object-based tasks, but at the cost of category-based tasks (especially MV category retrieval). This can be mitigated by adding $\mathcal{L}_{picat}$ that enhances performance on category-based tasks. However, $\mathcal{L}_{picat}$ and $\mathcal{L}_{piobj}$ have conflicting objectives in the same space and only marginally improve overall performance over $\mathcal{L}_{cat}$ in the single embedding space. In the dual embedding space, these losses are optimized in separate embedding spaces. In the dual space, we observe that $\mathcal{L}_{cat}+\mathcal{L}_{piobj}$ improves overall performance, particularly for object-based tasks, and adding $\mathcal{L}_{picat}$ boosts performance on category-based tasks and yields the best overall performance for all the datasets. $\mathcal{L}_{piobj}$ enhances the separability of object-identity classes facilitating learning more discriminative object embeddings that significantly improves performance on object-based tasks (see detailed ablation study of $\mathcal{L}_{piobj}$ in Sup. Sec 7).

**Optimizing Intra-class and Inter-class Distances:** In the top of Fig. 6, we show the maximum intra-class distance ($d^{max}_{intra}$) and minimum inter-class distance ($d^{min}_{inter}$) between object-identity classes from the same category during training on the ModelNet40 dataset. These distances are computed using the object-identity embeddings and averaged over all objects. We also plot the ratio $\rho = \frac{d^{min}_{inter}}{d^{max}_{intra}}$ in the bottom of Fig. 6. A higher $\rho$ value indicates embeddings of the same object-identity class are well clustered and separated from embeddings of other object-identity classes from the same category. Comparing the plots for the single and dual embedding spaces, we observe that $\rho$ and $d^{min}_{inter}$ are much higher for the dual space indicating better separability of object-identity classes when learning a dual space. We observe the same effect for all datasets (see Sup. Sec 8).

**Embedding Dimensionality:** In Fig. 7, we observe that for ModelNet-40, a dimension of 64 for category and 128 for object-level tasks is sufficient for good performance. For ObjectPI, higher dimensions of 256 and 512 are required for category and object-level tasks respectively to capture color and texture information in addition to shape, unlike ModelNet-40. A higher embedding dimensionality is required for object-level tasks compared to category-level tasks possibly because object embeddings need to capture finer details to effectively distinguish between objects. We provide more details in the Supplemental Sec. 9.

**Qualitative Results:** In the Supplemental, we illustrate how self-attention captures correlations between different views of an object using multi-view attention maps in Sec. 10, and present qualitative object retrieval results in Sec. 11.




*(a) ModelNet-40 dataset* *(b) ObjectPI dataset*

**Figure 7.** *Effect of embedding dimensionality on performance.*

## 5. Conclusion

We propose a multi-view dual-encoder architecture and pose-invariant ranking losses that facilitate learning discriminative pose-invariant representations for joint category and object recognition and retrieval. Our method outperforms state-of-the-art methods on several pose-invariant classification and retrieval tasks on three publicly available multi-view object datasets. We further provide ablation studies to demonstrate the effectiveness of our approach.

## References


[1] W. Chen, X. Chen, J. Zhang, and K. Huang. Beyond triplet loss: A deep quadruplet network for person re-identification. In *Computer Vision and Pattern Recognition (CVPR)*, pages 1320–1329, 2017. 2

[2] S. Chopra, R. Hadsell, and Y. LeCun. Learning a similarity metric discriminatively, with application to face verification. In *Computer Vision and Pattern Recognition (CVPR)*, pages 539–546 vol. 1, 2005. 2

[3] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations (ICLR)*, 2021. 3

[4] Weifeng Ge, Weilin Huang, Dengke Dong, and Matthew R. Scott. Deep metric learning with hierarchical triplet loss. In *European Conference on Computer Vision (ECCV)*, pages 272–288. Springer International Publishing, 2018. 2

[5] B. Harwood, V. Kumar B.G., G. Carneiro, I. Reid, and T. Drummond. Smart mining for deep metric learning. In *International Conference on Computer Vision (ICCV)*, pages 2840–2848, Los Alamitos, CA, USA, 2017. IEEE Computer Society. 2

[6] Xinwei He, Yang Zhou, Zhichao Zhou, Song Bai, and Xiang Bai. Triplet-center loss for multi-view 3d object retrieval. *Computer Vision and Pattern Recognition (CVPR)*, 2018. 2, 6

[7] Chih-Hui Ho, Pedro Morgado, Amir Persekian, and Nuno Vasconcelos. Pies: Pose invariant embeddings. In *Computer Vision and Pattern Recognition (CVPR)*, 2019. 1, 2, 3, 5, 6, 7, 10, 11, 12, 17

[8] Elad Hoffer and Nir Ailon. Deep metric learning using triplet network. *Lecture Notes in Computer Science*, page 84–92, 2015. 2

[9] Chen Huang, Chen Change Loy, and Xiaoou Tang. Local similarity-aware deep feature embedding. In *Neural Information Processing Systems (NeurIPS)*, page 1270–1278, Red Hook, NY, USA, 2016. Curran Associates Inc. 2

[10] Asako Kanezaki, Yasuyuki Matsushita, and Yoshifumi Nishida. Rotationnet: Joint object categorization and pose estimation using multiviews from unsupervised viewpoints. *Computer Vision and Pattern Recognition (CVPR)*, pages 5010–5019, 2018. 2

[11] Weiyang Liu, Yandong Wen, Zhiding Yu, and Meng Yang. Large-margin softmax loss for convolutional neural networks. In *International Conference on Machine Learning (ICML)*, page 507–516. JMLR.org, 2016. 5

[12] Xinhai Liu, Zhizhong Han, Yu-Shen Liu, and Matthias Zwicker. Fine-grained 3d shape classification with hierarchical part-view attentions. *IEEE Transactions on Image Processing*, 2021. 6

[13] Leland McInnes, John Healy, Nathaniel Saul, and Lukas Großberger. Umap: Uniform manifold approximation and projection. *Journal of Open Source Software*, 3(29):861, 2018. 7

[14] Yair Movshovitz-Attias, Alexander Toshev, Thomas K Leung, Sergey Ioffe, and Saurabh Singh. No fuss distance metric learning using proxies. In *International Conference on Computer Vision (ICCV)*, pages 360–368, 2017. 2

[15] Weizhi Nie, Yue Zhao, Dan Song, and Yue Gao. Dan: Deep-attention network for 3d shape recognition. *IEEE Trans. on Image Processing*, 30:4371–4383, 2021. 3

[16] Omkar M. Parkhi, Andrea Vedaldi, and Andrew Zisserman. Deep face recognition. In *British Machine Vision Conference (BMVC)*, pages 41.1–41.12. BMVA Press, 2015. 2

[17] Rohan Sarkar, Navaneeth Bodla, Mariya Vasileva, Yen-Liang Lin, Anurag Beniwal, Alan Lu, and Gerard Medioni. Outfittransformer: Outfit representations for fashion recommendation. In *Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 2263–2267, 2022. 3

[18] Rohan Sarkar, Navaneeth Bodla, Mariya I. Vasileva, Yen-Liang Lin, Anurag Beniwal, Alan Lu, and Gerard Medioni. Outfittransformer: Learning outfit representations for fashion recommendation. In *Winter Conference on Applications of Computer Vision (WACV)*, pages 3601–3609, 2023. 3

[19] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. *Computer Vision and Pattern Recognition (CVPR)*, 2015. 2

[20] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations (ICLR)*, 2015. 6

[21] Hyun Oh Song, Yu Xiang, Stefanie Jegelka, and Silvio Savarese. Deep metric learning via lifted structured feature embedding. In *Computer Vision and Pattern Recognition (CVPR)*, pages 4004–4012, 2016. 2

[22] Hang Su, Subhransu Maji, Evangelos Kalogerakis, and Erik G. Learned-Miller. Multi-view convolutional neural networks for 3d shape recognition. In *International Conference on Computer Vision (ICCV)*, 2015. 2, 6

[23] Yumin Suh, Bohyung Han, Wonsik Kim, and Kyoung Mu Lee. Stochastic class-based hard example mining for deep metric learning. In *Computer Vision and Pattern Recognition (CVPR)*, pages 7244–7252, 2019. 2

[24] Xun Wang, Haozhi Zhang, Weilin Huang, and Matthew R Scott. Cross-batch memory for embedding learning. In *Computer Vision and Pattern Recognition (CVPR)*, pages 6388–6397, 2020. 2

[25] X. Wei, Y. Gong, F. Wang, X. Sun, and J. Sun. Learning canonical view representation for 3d shape recognition with arbitrary views. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 397–406, Los Alamitos, CA, USA, 2021. IEEE Computer Society. 3

[26] Zhirong Wu, S. Song, A. Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and J. Xiao. 3d shapenets: A deep representation for volumetric shapes. In *Computer Vision and Pattern Recognition (CVPR)*, pages 1912–1920, Los Alamitos, CA, USA, 2015. IEEE Computer Society. 2, 5

[27] Hong Xuan, Abby Stylianou, Xiaotong Liu, and Robert Pless. Hard negative examples are hard, but useful. In *European Conference on Computer Vision (ECCV)*, pages 126–142. Springer International Publishing, 2020. 2