Training Vision Transformers for Image Retrieval

Alaaeldin El-Nouby [1 2] **Natalia Neverova** [1] **Ivan Laptev** [2] **Hervé Jégou** [1]

Abstract

Transformers have shown outstanding results for natural language understanding and, more recently, for image classification. We here extend this work and propose a transformer-based approach for image retrieval: we adopt vision transformers for generating image descriptors and train the resulting model with a metric learning objective, which combines a contrastive loss with a differential entropy regularizer.

Our results show consistent and significant improvements of transformers over convolution-based approaches. In particular, our method outperforms the state of the art on several public benchmarks for category-level retrieval, namely Stanford Online Product, In-Shop and CUB-200. Furthermore, our experiments on $\mathcal{R}$Oxford and $\mathcal{R}$Paris also show that, in comparable settings, transformers are competitive for particular object retrieval, especially in the regime of short vector representations and low-resolution images.

1. Introduction

One of the fundamental skills in reasoning is the ability to predict similarity between entities even if such entities have not been observed before. In the context of computer vision, learning similarity metric has many direct applications such as content-based image retrieval, face recognition and person re-identification. It is also a key component of many other computer vision tasks like zero-shot and few-shot learning. More recently, advances in metric learning have been essential to the progress of self-supervised learning, which relies on matching two images up to data augmentation as a learning paradigm.

Modern methods for image retrieval typically rely on convolutional encoders and extract compact image-level descriptors. Some early approaches used activations provided by off-the-shelf pre-trained models (Babenko et al., 2014). However, models trained specifically for the image retrieval task achieve better performance (Radenović et al., 2018b; Teh et al., 2020; Wang et al., 2020). A number of efficient objective functions have been proposed to penalize the discrepancy between computed similarities and the ground truth. In addition, research has been focused on improvements of sampling methods and data augmentation strategies.

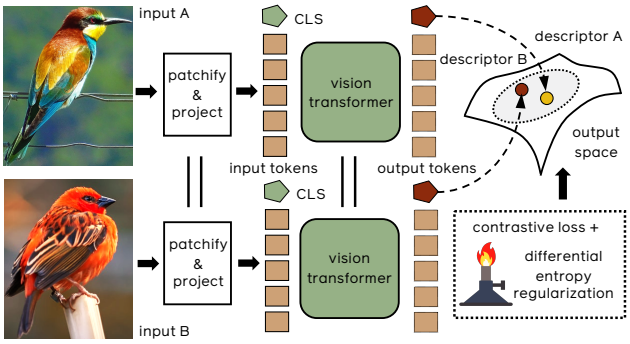


Figure 1. We train a transformer model with a Siamese architecture for image retrieval. Two input images are mapped by the transformers into a common feature space. At training time, the contrastive loss is augmented with an entropy regularizer.

The transformer architecture by Vaswani et al. (2017) has been successfully used for a number of NLP tasks (Devlin et al., 2018; Radford et al., 2018), and more recently in the core computer vision task of image classification (Dosovitskiy et al., 2021; Touvron et al., 2020). This is an interesting development as transformer-based models adapted for computer vision come with a different set of inductive biases compared to the currently dominant convolutional architectures. This suggests that such models may find alternative solutions and avoid errors that are typical for convolutional backbones. While there have been some efforts exploring attention-based metric learning for images (Kim et al., 2018; Chen & Deng, 2019), to our knowledge the adoption of a plain transformer has not been studied in this context.

In this work, we introduce and study *Image Retrieval Transformers* (IRT). As illustrated in Figure 1, our IRT model instantiates a Siamese architecture with a transformer backbone. We investigate the adaptation of metric learning techniques and evaluate how they interplay with transformers. In particular, we adopt a contrastive loss (Hadsell et al.,

[1]Facebook AI [2]ENS/Inria. Correspondence to: Alaaeldin El-Nouby <aelnouby@fb.com>.

2006), which has recently been reaffirmed as a very effective metric learning objective (Musgrave et al., 2020; Wang et al., 2020). We also employ a differential entropy regularization that favors uniformity over the representation space and improves performance.

We perform an extensive experimental evaluation and validate our approach by considering two image retrieval tasks. First, we investigate the task of category-level image retrieval, which is often used to measure the progress in metric learning (Teh et al., 2020; Musgrave et al., 2020). We also explore retrieval of particular objects, and compare our method to convolutional baselines in similar settings (same resolution and similar complexity).

The main contributions of this work are listed below.

- We propose a simple way to train vision transformers both for category-based level and particular object retrieval, and achieve competitive performance when compared to convolutional models with similar capacity.
- As a result, we establish the new state of the art on three popular benchmarks for category-level retrieval.
- For particular object retrieval, in the regime of short-vector representation (128 components), our results on $\mathcal{R}$Oxford and $\mathcal{R}$Paris are comparable to those of convnets operating at a much higher resolution and FLOPS.
- We show that the differential entropy regularizer enhances the contrastive loss and improves the performance overall.

2. Related Work

Transformers. The transformer architecture was introduced by Vaswani et al. (2017) for machine translation. It solely relies on self-attention and fully-connected layers, and achieving an attractive trade-off between efficiency and performance. It has subsequently provided state-of-the-art performance for several NLP tasks (Devlin et al., 2018; Radford et al., 2018). In computer vision, several attempts have been devoted to incorporate various forms of attention, for instance in conjunction (Wang et al., 2018), as a replacement to convolution (Ramachandran et al., 2019) or . Other methods utilize transformer layers on top of convolutional trunks (Carion et al., 2020) for detection.

More recently, convolution-free models that only rely on transformer layers have shown competitive performance (Chen et al., 2020; Dosovitskiy et al., 2021; Touvron et al., 2020), positioning it as a possible alternative to convolutional architectures. In particular, the Vision Transformers (ViT) model proposed by Dosovitskiy et al. (2021) is the first example of a transformer-based method to match or even surpass state-of-the-art convolutional models on the task of image classification. Touvron et al. (2020) subsequently improved the optimization procedure, leading to competitive results with ImageNet-only training (Deng et al., 2009).

Metric Learning. A first class of deep metric learning methods is based on classification: these approaches represent each category using one (Movshovitz-Attias et al., 2017; Teh et al., 2020; Zhai & Wu, 2018; Boudiaf et al., 2020) or multiple prototypes (Qian et al., 2019). The similarity and dissimilarity training signal is computed against the prototypes rather than between individual instances. Another class of methods operate on pairs methods: the training signal is defined by similarity/dissimilarity between individual instances directly. A contrastive loss (Hadsell et al., 2006) aims to push representations of positive pairs closer together, while representations of negative pairs are encouraged to have larger distance. The triplet loss (Weinberger & Saul, 2009) builds on the same idea but requires the positive pair to be closer than a negative pair by a fixed margin given the same anchor. Wu et al. (2017) proposes negative sampling weighted by pair-wise distance to emphasize harder negative examples. Other pair-based losses rely on the softmax function (Goldberger et al., 2005; Sohn, 2016; Wu et al., 2018; Wang et al., 2019), allowing for more comparisons between different positive and negative pairs.

While a vanilla contrastive loss has been regarded to have a weaker performance when compared to its successors like triplet and margin (Wu et al., 2017) losses, recent efforts (Musgrave et al., 2020) showed that a careful implementation of the contrastive loss leads to results outperforming many more sophisticated losses. Additionally, Wang et al. (2020) showed that when augmented with an external memory to allow sampling of a sufficient number of hard negatives, contrastive loss achieves a state-of-the-art performance on multiple image retrieval benchmarks.

Particular Image Retrieval has progressively evolved from methods based on local descriptors to convolutional encoders. In this context, an important design choice is how to compress the spatial feature maps of activations into a vector-shaped descriptor (Babenko & Lempitsky, 2015; Tolias et al., 2015). Subsequent works have adopted end-to-end training (Gordo et al., 2016; Radenović et al., 2018b; Revaud et al., 2019) with various forms of supervision. In a concurrent work, Gkelios et al. (2021) investigated off-the-shelf pre-trained ViT models for particular image retrieval.

Differential Entropy Regularization. Zhang et al. (2017) aim a better utilization of the space by spreading out the descriptors through matching first and second moments of non-matching pairs with points uniformly sampled on the sphere. Wang & Isola (2020) provide a theoretical analysis for contrastive representation learning in terms of alignment and uniformity on the hypersphere. In the context of face recognition, Duan et al. (2019) argue for spreading the class centers uniformly in the manifold, while Zhao et al. (2019) minimize the angle between a class center and its nearest neighbor in order to improve inter-class separability.

We focus our study on pairwise losses, the contrastive loss in particular, aiming to prevent the collapse in dimensions that happens as a byproduct of adopting such an objective. Most related to out method, Sablayrolles et al. (2019) propose a differential entropy regularization based on the estimator by Kozachenko & Leonenko (1987), in order to spread the vectors on the hypersphere more uniformly, such that it enables improved lattice-based quantization properties. Bell et al. (2020) adopted it as an efficient way to binarize features output by convnets in commerce applications.

3. Methods

In this section, after reviewing the transformer architecture, we detail how we adapt it to the category-level and particular object retrieval. Note, in the literature these tasks have been tackled by distinct techniques. In our case we use the same approach for both of these problems. We gradually introduce its different components, as follows:

- **IRT$_\mathbf{O}$** – off-the-shelf extraction of features from a ViT backbone, pre-trained on ImageNet;
- **IRT$_\mathbf{L}$** – fine-tuning a transformer with metric learning, in particular with a contrastive loss;
- **IRT$_\mathbf{R}$** – additionally regularizing the output feature space to encourage uniformity.

3.1. Preliminaries: Vision Transformer

Let us review the main building blocks for transformer-based models, and more specifically of the recently proposed ViT architecture by Dosovitskiy et al. (2021). The input image is first decomposed into M fixed-sized patches (e.g. 16×16). Each patch is linearly projected into M vector-shaped tokens and used as an input to the transformer in a permutation-invariant manner. The location prior is incorporated by adding a learnable 1-D positional encoding vector to the input tokens. An extra learnable CLS token is added to the input sequence such that its corresponding output token serves as a global image representation.

The transformer consists of L layers, each of which is composed of two main blocks: a Multi-Headed Self Attention (MSA) layer, which applies a self-attention operation to different projections of the input tokens, and a Feed-Forward Network (FFN). Both the MSA and FFN layers are preceded by layer normalization and followed by a skip connection. We refer the reader to Dosovitskiy et al. (2021) for details.

Architectures. Table 1 presents the neural networks models used through this paper. They are all pre-trained on ImageNet1k (Deng et al., 2009) only. In order to have a fair comparison with other retrieval methods, we choose to use the DeiT-Small variant of the ViT architecture introduced by Touvron et al. (2020) as our primary model. The DeiT-Small model has a relatively compact size which makes it comparable to the widely adopted ResNet-50 convolutional model (He et al., 2016) in terms of parameters count and FLOPS, as shown in Table 1. Additionally, we provide some analysis and results of larger models like ResNet-101 and DeiT-Base, as well as DeiT variants with advanced pre-training.

Table 1. Parameters count, FLOPS and Top-1 accuracy (%) on Imagenet1k-val for convolutional baselines ResNet-50 (R50) and ResNet-101 (R101) at resolution 224x224, as well as transformer-based models: DeiT-Small (DeiT-S) and DeiT-Base (DeiT-B) (Touvron et al., 2020). †*: Models pre-trained with distillation with a convnet trained on ImageNet1k.*

Model	# params	FLOPS (G)	Top-1 (%)
R50	23M	8.3	76.2
DeiT-S	22M	8.4	**79.8**
DeiT-S†	22M	8.5	**81.1**
R101	46M	15.7	77.4
DeiT-B	87M	33.7	**81.8**
DeiT-B†	87M	33.8	**83.9**

3.2. IRT$_\mathbf{O}$: off-the-shelf features with Transformers

We first consider the naive approach **IRT$_\mathbf{O}$**, where we extract features directly from a transformer pre-trained on ImageNet. This strategy is in line with early works on image retrieval with convolutional networks (Babenko et al., 2014), which were featurizing activations.

Pooling. We extract a compact vector descriptor that represents the image globally. In the ViT architecture, pre-classification layers output $M+1$ vectors corresponding to M input patches and a class (CLS) embedding.

In our referent pooling approach, **CLS**, we follow the spirit of BERT (Devlin et al., 2018) and ViT models, and view this class embedding as a global image descriptor. In addition, we investigate performance of global pooling methods that are typically used by convolutional metric learning models, including average, maximum and Generalized Mean (GeM) pooling, and apply them to the M output tokens.

l_2-Normalization and Dimensionality Reduction. We follow the common practice of projecting the descriptor vector into a unit ball after pooling. In the case when the target dimensionality is smaller than that provided by the architecture, we optionally reduce the vector by principal component analysis (PCA) before normalizing it.

3.3. IRT$_\mathbf{L}$: Learning the Metric for Image Retrieval

We now consider a metric learning approach for image retrieval, denoted by **IRT$_\mathbf{L}$**. It is the dominant approach to both category-level and particular object retrieval. In our case we combine it with transformers instead of convolutional neural networks. We adopt the contrastive loss with cross-batch memory by Wang et al. (2020) and fix the margin $\beta = 0.5$ by default for our metric learning objective.

The contrastive loss maximizes the similarity between encoded low-dimensional representations z_i of samples with the same label y (or any other pre-defined similarity rule). Simultaneously, it minimizes the similarity between representations of samples with unmatched labels which are referred to as negatives. For the contrastive loss, only negative pairs with a similarity higher than a constant margin β contribute to the loss. This prevents the training signal from being overwhelmed by easy negatives. Formally, the contrastive loss over a batch of size N is defined as:

$$\mathcal{L}_{\text{contr.}} = \frac{1}{N}\sum_{i}^{N}\left[\sum_{j:y_i=y_j}\left[1 - z_i^T z_j\right] + \sum_{j:y_i\neq y_j}\left[z_i^T z_j - \beta\right]_+\right]. \quad (1)$$

The representations z_i are assumed to be l_2-normalized, therefore the inner product is equivalent to cosine similarity.

3.4. IRT$_\mathbf{R}$: Differential Entropy Regularization

Recently, Boudiaf et al. (2020) studied connections between a group of pairwise losses and maximization of mutual information between learned representations $Z = \{z_i\}$ and corresponding ground-truth labels $Y = \{y_i\}$. We are interested in the particular case of the contrastive loss. The mutual information is defined as

$$\mathcal{I}(Z, Y) = \mathcal{H}(Z) - \mathcal{H}(Z|Y). \quad (2)$$

The positive term of the contrastive loss leads to minimization of the conditional differential entropy $\mathcal{H}(Z|Y)$, where intuitively, samples representations belonging to the same category are trained to be more similar:

$$\mathcal{H}(Z|Y) \propto \frac{1}{N}\sum_{i}^{N}\sum_{j:y_i=y_j}\left[1 - z_i^T z_j\right]. \quad (3)$$

On the other hand, the negative term of this loss is responsible for preventing trivial solutions where all sample representations are collapsed to a single point. Therefore, it maximizes the entropy of the learned representations:

$$\mathcal{H}(Z) \propto -\frac{1}{N}\sum_{i}^{N}\sum_{j:y_i\neq y_j}[z_i^T z_j - \beta]_+. \quad (4)$$

The margin β plays an important role in the training dynamics. Low values of β allow exploration of a larger number of negative samples. Yet in this case easy negatives can dominate the training and cause the performance to plateau. In contrast, higher values of β would only accept hard negatives, possibly leading to noisy gradients and unstable training (Wu et al., 2017).

Our regularizer. Motivated by the entropy maximization view of the negative contrastive term in Equation 4, we add an entropy maximization term that is independent of the negative samples accepted by the margin. In particular, we use the differential entropy loss proposed by Sablayrolles et al. (2019). It is based on the Kozachenko & Leonenko (1987) differential entropy estimator:

$$\mathcal{L}_{\text{KoLeo}} = -\frac{1}{N}\sum_{i}^{N}\log(\rho_i), \quad (5)$$

where $\rho_i = \min_{i\neq j}\|z_i - z_j\|$. In other words, this regularization maximizes the distance between every point and its nearest neighbor, and therefore alleviates the collapse issue. We simply add the regularization term to the contrastive loss weighted by a regularization strength coefficient λ: $\mathcal{L} = \mathcal{L}_{\text{contr.}} + \lambda\mathcal{L}_{\text{KoLeo}}$.

Intuitively, the different entropy regularization prevents the representations of different samples from lying too close on the hypersphere, by increasing their distance from positive examples, and the hard negatives as well. Having hard negatives with extremely small distances is a main source of noise in the training signal, as identified by Wu et al. (2017).

3.5. Analysis

We study the behaviour of the output representation space when training with a contrastive loss, and how augmenting this loss with a differential entropy regularization impacts the space properties and the model performance.

PCA in the Embedding Space. In Figure 2 we examine the cumulative energy of the principle components for features from an off-the-shelf, ImageNet pre-trained model, as well as models trained using contrastive loss. We observe that the features after training with the contrastive loss suffer from a collapse in dimensions compared to an untrained model. This suggests an ineffective use of the representational capacity of the embedding space, as alignment is favored over uniformity while both are necessary for good representations (Wang & Isola, 2020). As we augment the contrastive loss with the differential entropy regularization, the cumulative energy spreads across more dimensions (see Figure 2 with non-zero values of λ). Higher values of λ alleviate the dimensionality collapse problem.

Another observation is that the transformer-based architecture is less impacted than convnets by the collapse (see Figure 3). Despite having a lower extrinsic dimensionality compared to the ResNet-50 model, the DeiT-Small features are more spread over principle components. A possible reason for that behavior is that in multi-headed attention, each input feature is projected to different sub-spaces before the attention operation, reducing the risk of collapse.

Gradient Analysis. As pointed out by Wu et al. (2017), very hard negatives can lead to noisy gradients. We examine the nuclear norm $\|\cdot\|_*$ associated with the covariance matrix of the gradients directions $\gamma = \|\text{Cov}(\nabla_z\mathcal{L}_{\text{contr.}})\|_*$, averaged over all training iterations (see Figure 4). Higher values of γ could indicate noisy gradients. We observe them for both very high and very low values of margin β which

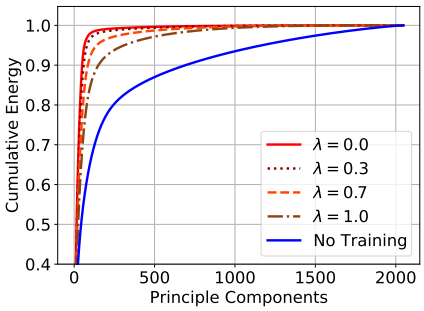


Figure 2. Cumulative energy of the principle components for features extracted using a ResNet-50 backbone from the SOP dataset, with pre-training on ImageNet, with (red) or without (blue) finetuning. The solid red line indicates the vanilla contrastive loss with $\beta = 0.5$. The features have collapsed to few dimensions after training, but the collapse is reduced by entropy regularization.

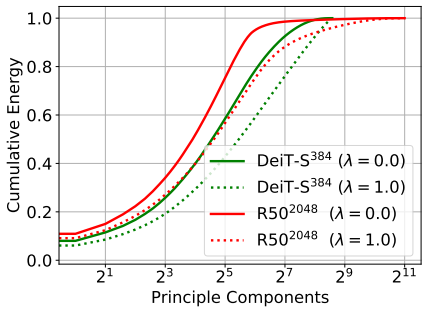


Figure 3. Cumulative energy of the principle components for ResNet-50 and DeiT-Small models trained using a margin of $\beta = 0.5$. We can see that despite the lower extrinsic dimensionality of DeiT-S ($384 \approx 2^{8.5}$), it has higher intrinsic dimensionalities than ResNet-50 after training. This suggests that the transformer-based architectures can be more robust against the feature collapse.

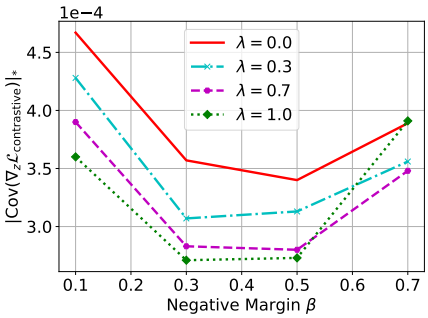


Figure 4. The nuclear norm for the covariance of the gradient direction, computed for different margin values β and entropy regularization strengths λ. The gradient signal is noisier for very high and very low margin values. The $\mathcal{L}_{\text{KoLeo}}$ regularization provides a more stable gradient signal, but, in turn, becomes harmful with high values of λ (e.g. $\lambda = 1.0, \beta = 0.7$).

aligns with our understanding that very easy and very hard negatives lead to less informative and less stable training signal. Moreover, we observe a decrease in the γ values after the addition of the entropy regularization term.

4. Experiments & Ablation Studies

We first describe datasets and implementation details, and then proceed with discussions of empirical results.

4.1. Datasets

Category-level Retrieval. We report performance on three popular datasets commonly used for category-level retrieval. **Stanford Online Products** (SOP) (Oh Song et al., 2016) consists of online products images representing 22,634 categories. Following the split proposed by Oh Song et al. (2016), we use first 11,318 categories for training and the remaining 11,316 for testing. **CUB-200-2011** (Wah et al., 2011) contains 11,788 images corresponding to 200 bird categories. Following Wah et al. (2011), we split this dataset into two class-disjoint sets, each with 100 categories for training and testing. **In-Shop** (Liu et al., 2016) contains 72,712 images of clothing items belonging to 7,986 categories, 3,997 of which used for training. The remaining 3,985 categories are split into 14,218 query and 12,612 gallery images for testing. We compute the Recall@K evaluation metric for a direct comparison with previous methods.

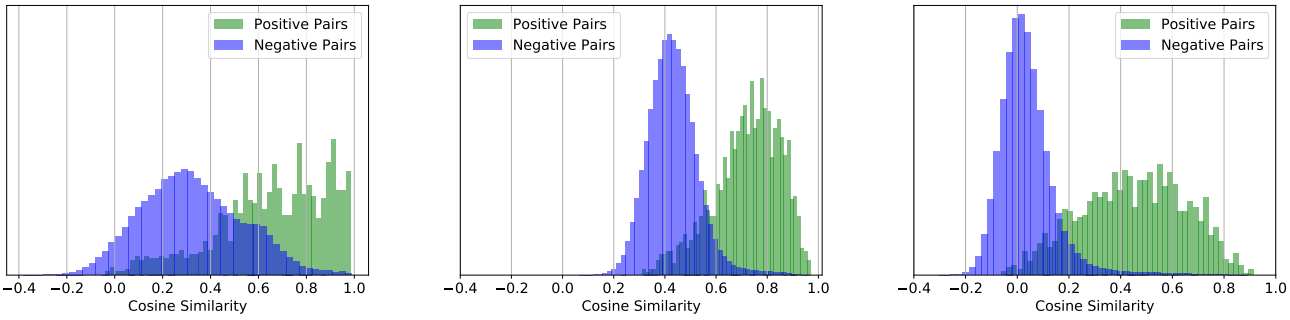
Particular Object Retrieval. For training, we use the **SFM120k** dataset (Radenović et al., 2018b) which is obtained by applying structure-from-motion and 3D reconstruction to large unlabelled image collections (Schonberger et al., 2015). The positive images are selected such that enough 3D points are co-observed with the query image, while negative images come from different 3D models. We use 551 3D models for training and 162 for validation.

For evaluation, we report results using revisited benchmarks (Radenović et al., 2018a) of the **Oxford** and **Paris** (Philbin et al., 2007; 2008) datasets. These two datasets each contain 70 query images depicting buildings, and additionally include 4993 and 6322 images respectively in which the same query buildings may appear. The revisited benchmarks contain 3 splits: Easy (E), Medium (M) and Hard (H), grouped by gradual difficulty of query/database pairs. (E) ignores hard queries, (M) includes both easy and hard ones, while (H) considers hard queries only. We report the Mean Average Precision (mAP) for the Medium and Hard splits in all our experiments.

4.2. Implementation & Training Details

Category-level Retrieval. The transformer-based models and their pre-trained weights are based on the public implementation[1] of DeiT (Touvron et al., 2020) built upon the Timm library by Wightman (2019). All models are optimized using the AdamW optimizer (Loshchilov & Hutter, 2017) with learning rate 3.10^{-5}, weight decay 5.10^{-4} and batch size of 64. For all experiments, unless mentioned otherwise, the contrastive loss margin is set to $\beta = 0.5$ and the entropy regularization strength is set to $\lambda = 0.7$. We show later in ablation that the results are relatively stable (and not overfitted) to this hyper-parameter setting. We use standard data augmentation methods of resizing the image to 256×256 and then taking a random crop of size 224×224,

[1] https://github.com/facebookresearch/deit



(a) $\mathbf{IRT_O}$: off-the-shelf descriptors. (b) $\mathbf{IRT_L}$: finetuned descriptors. (c) $\mathbf{IRT_R}$: finetuned with $\lambda = 4.0$.

Figure 5. We present histograms for the cosine similarities belonging to positive and negative pairs for three DeiT-Small variants. The descriptors are extracted from the $\mathcal{R}$Paris (M) dataset with input images of size 224×224 and descriptor dimensionality of 256-d for all models. Extracting features using off-the-shelf features (left) results in a highly overlapping positive and negative distribution. Finetuning the model using contrastive loss (middle) reduces the overlap significantly. However, the descriptors concentrate relatively close to each other, not taking full advantage of the embedding space. Augmenting the contrastive loss with entropy regularization (right) results in a more uniformly spread distribution of descriptors and a better separation of positive and negative pairs based on absolute similarities.

combined with random horizontal flipping. Following Wang et al. (2020), we use a dynamic offline memory queue of the same size as the dataset (with the exception of In-Shop dataset for which the memory size is 0.2 of the dataset size). Additionally, for the In-Shop dataset we adopt a momentum encoder for the memory entries (similarly to He et al. (2020)) with momentum value of 0.999. We have found this was not necessary for SOP and CUB-200-2011. Finally, SOP and In-Shop models were trained for 35k iterations and the CUB-200-2011 model was trained for 2000 iterations.

Particular Object Retrieval. For the particular object retrieval experiments, we build our implementation on top of the public code[2] associated with the work by Radenović et al. (2018b). We follow the same optimization and regularization procedure. All models, transformer-based and convolutional, are finetuned using the SFM120k dataset. The input images are resized to have the same fixed width and height. We report results for image sizes of 224×224 and 384×384. For finetuning, each batch consists of 5 tuples of (1 anchor, 1 positive, 5 negatives). For each epoch, we randomly select 2,000 positive pairs and 22,000 negative candidates (using hard-negative mining). We use the default hyper-parameters of Radenović et al. (2018b): the models are optimized using Adam (Kingma & Ba, 2015) with small learning rate of 5.10^{-7} and weight decay of 10^{-6}. The contrastive loss margin is set to $\beta = 0.85$. The models are finetuned for 100 epochs. All models with GeM pooling use a pooling exponent value of $p = 3$. The dimensionality reduction is achieved using a PCA trained on the SFM120k dataset. For the evaluation, all the query and database images are resized into a square image with the same resolution as used during the finetuning stage.

[2]https://github.com/filipradenovic/cnnimageretrieval-pytorch

4.3. Results

Category-level Retrieval. We present the Recall@K performance for three public benchmarks for category-level retrieval. For the SOP dataset, we can see in Table 2 that our $\mathbf{IRT_R}$ model with DeiT-S^{384} backbone achieves state-of-the-art performance for all values of K, outperforming previous methods by a margin of 2.6% absolute points for Recall@1. The DeiT-S† variant with distillation pre-training achieves the best results on this benchmark. Even when reducing the dimensionality to 128-D, our method outperforms all the convnets except at Recall@1000. On the CUB-200-2011 dataset, the DeiT-S^{384} model outperforms the current state of the art by 2.5% points at Recall@1. The distilled DeiT-S model provides an additional 1.9% improvement, achieving the best results for all values of K. The DeiT-S^{128} variant with compressed representation outperforms all previous methods except for the ProxyNCA++ model that uses 2048-D descriptors. Similarly, for In-Shop, the DeiT-S^{384} model and its distilled variant outperform all previous models at Recall@1 with a margin of 0.2% and 0.6% respectively.

Particular Object Retrieval. We present the mAP performance for the Medium and Hard splits of the revisited Oxford and Paris benchmarks in Table 7. First observe that for input images with size 224×224, the DeiT-S† backbone outperforms its ResNet-50 counterpart with the same capacity, as well as the higher capacity ResNet-101 across all benchmarks and descriptor sizes. The larger DeiT-B† provides a significant gain in performance and achieves the best result among the reported models. Scaling up the image size to 384×384 considerably improves the performance for all models with the DeiT-B† model retaining its position as the strongest model. In Table 8 we compare our model to strong state-of-the-art methods in particular object retrieval, following the standard extensive evaluation procedure. Revaud

Table 2. Recall@K performance for the SOP, CUB-200 and In-Shop category-level datasets compared to the state-of-the-art methods. ▷128: reduction to 128 components obtained using PCA.

Method	Backbone	#dims	SOP (K)				CUB-200 (K)				In-Shop (K)			
			1	10	100	1000	1	2	4	8	1	10	20	30
A-BIER (Opitz et al., 2018)	GoogleNet	512	74.2	86.9	94.0	97.8	57.5	68.7	78.3	86.2	83.1	95.1	96.9	97.5
ABE (Kim et al., 2018)			76.3	88.4	94.8	98.2	60.6	71.5	79.8	87.4	87.3	96.7	97.9	98.2
SM (Suh et al., 2019)			75.3	87.5	93.7	97.4	56.0	68.3	78.2	86.3	90.7	97.8	98.5	98.8
XBM (Wang et al., 2020)			77.4	89.6	95.4	98.4	61.9	72.9	81.2	88.6	89.4	97.5	98.3	98.6
HTL (Ge, 2018)	InceptionBN	512	74.8	88.3	94.8	98.4	57.1	68.8	78.7	86.5	80.9	94.3	95.8	97.2
MS (Wang et al., 2019)			78.2	90.5	96.0	98.7	65.7	77.0	86.3	91.2	89.7	97.9	98.5	98.8
SoftTriple (Qian et al., 2019)			78.6	86.6	91.8	95.4	65.4	76.4	84.5	90.4				
XBM (Wang et al., 2020)			79.5	90.8	96.1	98.7	65.8	75.9	84.0	89.9	89.9	97.6	98.4	98.6
HORDE (Jacob et al., 2019)			80.1	91.3	96.2	98.7	66.8	77.4	85.1	91.0	90.4	97.8	98.4	98.7
Margin (Wu et al., 2017)	ResNet-50	128	72.7	86.2	93.8	98.0	63.9	75.3	84.4	90.6	-	-	-	-
FastAP (Cakir et al., 2019)			73.8	88.0	94.9	98.3	-	-	-	-	-	-	-	-
MIC (Roth et al., 2019)			77.2	89.4	94.6	-	66.1	76.8	85.6	-	88.2	97.0	-	98.0
XBM (Wang et al., 2020)			80.6	91.6	96.2	98.7	-	-	-	-	91.3	97.8	98.4	98.7
NSoftmax (Zhai & Wu, 2018)		512	78.2	90.6	96.2	-	61.3	73.9	83.5	90.0	86.6	97.5	98.4	98.8
ProxyNCA++ (Teh et al., 2020)			80.7	92.0	96.7	98.9	69.0	79.8	87.3	92.7	90.4	98.1	98.8	99.0
NSoftmax (Zhai & Wu, 2018)		2048	79.5	91.5	96.7	-	65.3	76.7	85.4	91.8	89.4	97.8	98.7	99.0
ProxyNCA++ (Teh et al., 2020)			81.4	92.4	96.9	99.0	72.2	82.0	89.2	**93.5**	90.9	**98.2**	**98.9**	**99.1**
$\mathbf{IRT_R}$ (ours)	DeiT-S	▷128	83.4	93.0	97.0	99.0	72.6	81.9	88.7	92.8	91.1	98.1	98.6	99.0
		384	**84.0**	**93.6**	**97.2**	**99.1**	**74.7**	**82.9**	**89.3**	93.3	**91.5**	98.1	98.7	99.0
	DeiT-S†	384	**84.2**	**93.7**	**97.3**	**99.1**	**76.6**	**85.0**	**91.1**	**94.3**	**91.9**	98.1	98.7	98.9

Table 3. Ablation of model components: off-the-shelf performance ($\mathbf{IRT_O}$), with contrastive learning ($\mathbf{IRT_L}$) and finally regularized ($\mathbf{IRT_R}$). All methods use a DeiT-S backbone with #dims=384.

Supervision ↓	SOP	CUB	In-Shop	$\mathcal{R}$Ox		$\mathcal{R}$Par	
				M	H	M	H
$\mathbf{IRT_O}$	52.8	58.5	31.3	20.3	5.6	50.2	26.3
$\mathbf{IRT_L}$	83.0	74.2	90.3	32.7	11.4	63.6	37.8
$\mathbf{IRT_R}$	**84.0**	**74.7**	**91.5**	**34.0**	**11.5**	**66.1**	**40.2**

et al. (2019) use the original resolution of the dataset (i.e. 1024×768), while Radenović et al. (2018a) utilizes multi-scale evaluation. Although these methods outperform our DeiT-B† model at resolution 384×384 in mAP, they are approximately 248% and 437% more expensive w.r.t. FLOPS. Furthermore, we observe that for compressed representations of 128-D, our model closes the gap with Radenović et al. (2018a), achieving a higher mAP for $\mathcal{R}$Paris.

4.4. Ablations

Different Methods of Supervision. We provide a comparison between different degrees of supervision corresponding to $\mathbf{IRT_O}$, $\mathbf{IRT_L}$ and $\mathbf{IRT_R}$ in Table 3. We observe that finetuning substantially improves performance over off-the-shelf features, especially for category-level retrieval. Augmenting the contrastive loss with differential entropy regularization further improves the performance across all benchmarks. Figure 5 demonstrates how the distribution of the cosine similarities between positive and negative pairs is impacted by the different variants we study. We notice that finetuning strongly helps to make the positive and negative distributions more separable. The entropy regularization term spreads the similarity values across a wider range.

Choice of Feature Extractor: Pooling Methods. In Table 9, we study different feature aggregation methods, as described in Section 3.2. Both for category-level and particular object retrieval, we observe that utilizing the CLS token as the image-level descriptor provides the strongest performance (or at least on par) compared to other popular pooling methods such as average pooling, max pooling and GeM. This suggests that the transformer operates as a learned aggregation operator, thereby reducing the need for careful design of feature aggregation methods.

Performance across Objective Functions. The choice of the objective function used to train image descriptors is crucially important and is the focus of the majority of the metric learning research. While we adopt the contrastive loss as our primary objective function, we additionally investigate two objective functions with different properties: (1) Normalized Softmax (Zhai & Wu, 2018) as a classification-based objective, and (2) Scalable Neighborhood Component Analysis (NCA) (Wu et al., 2018), a pairwise objective with implicit weighting of hard negatives through temperature. Table 4 shows that DeiT-S outperforms its convolutional counterpart across all different choices of objective functions. This suggests that transformer-based models are strong metric learners and hence an attractive alternative to convolutional models for image retrieval.

Table 4. Comparison between convolutional ResNet-50 (R50) and $\mathbf{IRT_L}$ DeiT-Small (DeiT-S) architectures across multiple metric learning objective functions, as tested using the SOP dataset, $\beta = 0.5$ ("Contr." refers to the contrastive loss we use).

Model	Loss	#dims	R@1
R50	NSoftmax	2048	79.5
DeiT-S		384	**80.8**
R50	SNCA	2048	78.0
DeiT-S		384	**81.1**
R50	Contr.	2048	79.8
DeiT-S		384	**83.0**

Table 5. Recall@1 results for different values of margins β and entropy regularization strengths λ on SOP. Entropy regularization consistently boosts performance (drops again for $\lambda > 1.0$).

Model	β	λ				
		0.0	0.3	0.5	0.7	1.0
R50	0.1	78.9	**79.2**	79.1	78.8	78.8
	0.3	79.3	81.3	**81.7**	**81.7**	**81.7**
	0.5	79.8	80.7	81.0	81.2	**81.3**
	0.7	79.3	80.5	**81.2**	81.1	78.5
	0.9	79.1	**80.0**	31.4	13.4	9.1
DeiT-S	0.1	70.3	70.7	**71.4**	70.1	71.1
	0.3	82.5	83.0	83.0	**83.1**	**83.1**
	0.5	83.0	83.5	83.6	**84.0**	**84.0**
	0.7	82.9	83.8	84.1	**84.2**	83.8
	0.9	82.6	**84.4**	80.6	71.2	41.5

Table 6. Particular object retrieval mAP performance for different entropy regularization strengths λ. The results are obtained by using DeiT-Small model with CLS token as the feature descriptor (#dims=384, trained with contrastive loss).

Model	λ	$\mathcal{R}$O		$\mathcal{R}$Par	
		M	H	M	H
DeiT-S	0.0	32.7	11.4	63.6	37.8
	1.0	31.5	9.3	64.7	38.6
	2.0	34.5	11.1	65.7	39.8
	3.0	**34.6**	**11.5**	**66.1**	40.1
	4.0	34.0	**11.5**	**66.1**	**40.2**
	5.0	32.3	10.4	65.6	39.8

Table 7. Particular object retrieval mAP performance comparison between different convolutional and $\mathbf{IRT_L}$ models using different descriptor dimensions. All models are finetuned the same way. ▷128: reduction to 128 components obtained using PCA.

Input size	Model	Pooler	#dims	$\mathcal{R}$Ox		$\mathcal{R}$Par	
				M	H	M	H
224×224	R50	GeM	▷128	25.8	8.6	56.7	31.2
	R50	R-MAC		23.6	5.5	56.0	30.8
	R101	GeM		27.8	8.0	59.0	32.2
	R101	R-MAC		27.3	7.4	57.9	31.3
	DeiT-S†	CLS		32.1	13.3	63.8	**39.3**
	DeiT-B†	CLS		**36.6**	**14.8**	**64.4**	39.1
224×224	R50	GeM	2048	28.7	10.9	61.2	35.9
	R50	R-MAC		25.6	7.3	60.6	35.4
	R101	GeM		31.7	11.1	63.4	37.3
	R101	R-MAC		31.0	9.3	62.6	36.5
	DeiT-S†	CLS	384	34.5	15.8	65.8	42.0
	DeiT-B†	CLS	768	**39.5**	**17.4**	**67.5**	**43.6**
384×384	R101	GeM	▷128	34.1	9.5	62.6	36.3
	R101	R-MAC		31.4	7.4	61.6	35.3
	DeiT-B†	CLS		**49.0**	**21.5**	**68.5**	**43.8**
384×384	R101	GeM	2048	38.1	12.5	69.4	45.8
	R101	R-MAC		37.1	10.6	66.0	41.4
	DeiT-B†	CLS	768	**50.5**	**22.7**	**70.6**	**47.4**

Table 8. Comparison with SoA methods for particular object retrieval: [1] (Radenović et al., 2018a), [2] (Revaud et al., 2019). ▷128: reduction to 128 components obtained using PCA. ⋆: FLOPS (G) are computed for input images of size 1024×768. §: our evaluation using pre-trained models from the authors.

Method	Model {maxres}	#dims	FLOPS (G)	$\mathcal{R}$Ox		$\mathcal{R}$Par	
				M	H	M	H
$\mathbf{IRT_L}$	DeiT-B†{384}	▷128	98.8	49.0	21.5	68.5	43.8
$\mathbf{IRT_R}$				49.1	21.1	68.3	44.1
$\mathbf{IRT_L}$		768		50.5	22.7	70.6	47.4
$\mathbf{IRT_R}$				55.1	28.3	72.7	49.6
[1]-GeM§	R101{1024}	▷128	432.2⋆	53.2	28.9	65.4	36.9
[1]-GeM		2048		64.7	38.5	77.2	56.3
[2]-GeM	R101{1024}	2048	246.0⋆	67.2	42.8	80.1	60.5

Table 9. Performance of different pooling methods on both retrieval tasks ($\mathbf{IRT_L}$ model, DeiT-Small backbone, #dims=384).

Pooler	SOP	CUB	In-Shop	$\mathcal{R}$Ox		$\mathcal{R}$Par	
				M	H	M	H
Average Pool	**83.0**	72.8	90.2	28.3	8.5	61.9	36.0
Max Pool	82.2	69.2	90.3	25.2	6.8	60.4	34.1
GeM	82.6	69.1	89.8	26.5	8.5	60.2	33.7
CLS	**83.0**	**74.4**	**90.4**	**32.7**	**11.4**	**63.6**	**37.8**

Regularizing Hyper-parameter λ. We explore the differential entropy regularization strength and its impact on the improvement of retrieval performance. First, we use the SOP dataset for our analysis and show how the Recall@1 performance changes with different margin values β and entropy regularization strengths λ in Table 5.

All models, either transformer-based or convolutional, trained with different margins are improved by the $\mathcal{L}_{\mathrm{KoLeo}}$ regularizer. The margins with the best results are those with the lowest γ values in Figure 4. Moreover, we observe a similar boost in performance for particular object retrieval in Table 6, confirming that the differential entropy regularization provides a clear and consistent improvement across different tasks and architectures.

5. Conclusion

In this paper, we have explored how to adapt the transformer architecture to metric learning and image retrieval. In this context, we have revisited the contrastive loss formulation and showed that a regularizer based on a differential entropy loss spreading vectors over the unit hyper-sphere improves the performance for transformer-based models, as well as for convolutional models. As a result, we establish the new state of the art for category-level image retrieval. Finally, we demonstrated that, for comparable settings, transformer-based models are an attractive alternative to convolutional backbones for particular object retrieval, especially with short vector representations. Their performance is competitive against convnets having a much higher complexity.

Acknowledgement Alaaeldin El-Nouby was supported in part by a FAIR/Prairie CIFRE PhD Fellowship. Ivan Laptev was supported in part by the French government under management of Agence Nationale de la Recherche as part of the "Investissements d'avenir" program, reference ANR- 19-P3IA-0001 (PRAIRIE 3IA Institute) and the Louis Vuitton/ENS Chair in Artificial Intelligence.

References

Babenko, A. and Lempitsky, V. Aggregating deep convolutional features for image retrieval. *arXiv preprint arXiv:1510.07493*, 2015.

Babenko, A., Slesarev, A., Chigorin, A., and Lempitsky, V. Neural codes for image retrieval. In *European Conference on Computer Vision*, 2014.

Bell, S., Liu, Y., Alsheikh, S., Tang, Y., Pizzi, E., Henning, M., Singh, K., Parkhi, O., and Borisyuk, F. Groknet: Unified computer vision model trunk and embeddings for commerce. In *SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2020.

Boudiaf, M., Rony, J., Ziko, I. M., Granger, E., Pedersoli, M., Piantanida, P., and Ayed, I. B. A unifying mutual information view of metric learning: cross-entropy vs. pairwise losses. In *European Conference on Computer Vision*, 2020.

Cakir, F., He, K., Xia, X., Kulis, B., and Sclaroff, S. Deep metric learning to rank. In *Computer Vision and Pattern Recognition*, 2019.

Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., and Zagoruyko, S. End-to-end object detection with transformers. In *European Conference on Computer Vision*, 2020.

Chen, B. and Deng, W. Hybrid-attention based decoupled metric learning for zero-shot image retrieval. In *Computer Vision and Pattern Recognition*, 2019.

Chen, M., Radford, A., Child, R., Wu, J., Jun, H., Luan, D., and Sutskever, I. Generative pretraining from pixels. In *International Conference on Machine Learning*, 2020.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition*, 2009.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021.

Duan, Y., Lu, J., and Zhou, J. Uniformface: Learning deep equidistributed representation for face recognition. In *Computer Vision and Pattern Recognition*, 2019.

Ge, W. Deep metric learning with hierarchical triplet loss. In *European Conference on Computer Vision*, 2018.

Gkelios, S., Boutalis, Y., and Chatzichristofis, S. A. Investigating the vision transformer model for image retrieval tasks. *arXiv preprint arXiv:2101.03771*, 2021.

Goldberger, J., Hinton, G. E., Roweis, S., and Salakhutdinov, R. R. Neighbourhood components analysis. In Saul, L., Weiss, Y., and Bottou, L. (eds.), *Advances in Neural Information Processing Systems*. MIT Press, 2005.

Gordo, A., Almazán, J., Revaud, J., and Larlus, D. Deep image retrieval: Learning global representations for image search. In *European Conference on Computer Vision*, 2016.

Hadsell, R., Chopra, S., and LeCun, Y. Dimensionality reduction by learning an invariant mapping. In *Computer Vision and Pattern Recognition*, 2006.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Computer Vision and Pattern Recognition*, 2016.

He, K., Fan, H., Wu, Y., Xie, S., and Girshick, R. Momentum contrast for unsupervised visual representation learning. In *Computer Vision and Pattern Recognition*, 2020.

Jacob, P., Picard, D., Histace, A., and Klein, E. Metric learning with horde: High-order regularizer for deep embeddings. In *International Conference on Computer Vision*, 2019.

Kim, W., Goyal, B., Chawla, K., Lee, J., and Kwon, K. Attention-based ensemble for deep metric learning. In *European Conference on Computer Vision*, 2018.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015.

Kozachenko, L. and Leonenko, N. N. Sample estimate of the entropy of a random vector. *Problemy Peredachi Informatsii*, 1987.

Liu, Z., Luo, P., Qiu, S., Wang, X., and Tang, X. Deepfashion: Powering robust clothes recognition and retrieval with rich annotations. In *Computer Vision and Pattern Recognition*, 2016.

Loshchilov, I. and Hutter, F. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Movshovitz-Attias, Y., Toshev, A., Leung, T. K., Ioffe, S., and Singh, S. No fuss distance metric learning using proxies. In *International Conference on Computer Vision*,

2017.

Musgrave, K., Belongie, S., and Lim, S.-N. A metric learning reality check. *arXiv preprint arXiv:2003.08505*, 2020.

Oh Song, H., Xiang, Y., Jegelka, S., and Savarese, S. Deep metric learning via lifted structured feature embedding. In *Computer Vision and Pattern Recognition*, 2016.

Opitz, M., Waltner, G., Possegger, H., and Bischof, H. Deep metric learning with bier: Boosting independent embeddings robustly. IEEE *Transactions on Pattern Analysis and Machine Intelligence*, 2018.

Philbin, J., Chum, O., Isard, M., Sivic, J., and Zisserman, A. Object retrieval with large vocabularies and fast spatial matching. In *International Conference on Computer Vision*, 2007.

Philbin, J., Chum, O., Isard, M., Sivic, J., and Zisserman, A. Lost in quantization: Improving particular object retrieval in large scale image databases. In *Computer Vision and Pattern Recognition*, 2008.

Qian, Q., Shang, L., Sun, B., Hu, J., Li, H., and Jin, R. Softtriple loss: Deep metric learning without triplet sampling. In *Computer Vision and Pattern Recognition*, 2019.

Radenović, F., Iscen, A., Tolias, G., Avrithis, Y., and Chum, O. Revisiting oxford and paris: Large-scale image retrieval benchmarking. In *Computer Vision and Pattern Recognition*, 2018a.

Radenović, F., Tolias, G., and Chum, O. Fine-tuning cnn image retrieval with no human annotation. IEEE *Transactions on Pattern Analysis and Machine Intelligence*, 2018b.

Radford, A., Narasimhan, K., Salimans, T., and Sutskever, I. Improving language understanding with unsupervised learning. *Technical report, OpenAI*, 2018.

Ramachandran, P., Parmar, N., Vaswani, A., Bello, I., Levskaya, A., and Shlens, J. Stand-alone self-attention in vision models. *arXiv preprint arXiv:1906.05909*, 2019.

Revaud, J., Almazán, J., Rezende, R. S., and Souza, C. R. d. Learning with average precision: Training image retrieval with a listwise loss. In *International Conference on Computer Vision*, 2019.

Roth, K., Brattoli, B., and Ommer, B. Mic: Mining interclass characteristics for improved metric learning. In *International Conference on Computer Vision*, 2019.

Sablayrolles, A., Douze, M., Schmid, C., and Jégou, H. Spreading vectors for similarity search. In *International Conference on Learning Representations*, 2019.

Schonberger, J. L., Radenovic, F., Chum, O., and Frahm, J.-M. From single image query to detailed 3d reconstruction. In *International Conference on Computer Vision*, 2015.

Sohn, K. Improved deep metric learning with multi-class n-pair loss objective. In *Advances in Neural Information Processing Systems*, 2016.

Suh, Y., Han, B., Kim, W., and Lee, K. M. Stochastic class-based hard example mining for deep metric learning. In *Computer Vision and Pattern Recognition*, 2019.

Teh, E. W., DeVries, T., and Taylor, G. W. Proxynca++: Revisiting and revitalizing proxy neighborhood component analysis. *arXiv preprint arXiv:2004.01113*, 2020.

Tolias, G., Sicre, R., and Jégou, H. Particular object retrieval with integral max-pooling of cnn activations. *arXiv preprint arXiv:1511.05879*, 2015.

Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., and Jégou, H. Training data-efficient image transformers and distillation through attention. *arXiv preprint arXiv:2012.12877*, 2020.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. In *Advances in Neural Information Processing Systems*, 2017.

Wah, C., Branson, S., Welinder, P., Perona, P., and Belongie, S. The caltech-ucsd birds-200-2011 dataset. 2011.

Wang, T. and Isola, P. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. *arXiv preprint arXiv:2005.10242*, 2020.

Wang, X., Girshick, R., Gupta, A., and He, K. Non-local neural networks. In *Computer Vision and Pattern Recognition*, 2018.

Wang, X., Han, X., Huang, W., Dong, D., and Scott, M. R. Multi-similarity loss with general pair weighting for deep metric learning. In *Computer Vision and Pattern Recognition*, 2019.

Wang, X., Zhang, H., Huang, W., and Scott, M. R. Cross-batch memory for embedding learning. In *Computer Vision and Pattern Recognition*, 2020.

Weinberger, K. Q. and Saul, L. K. Distance metric learning for large margin nearest neighbor classification. *Journal of Machine Learning Research*, 2009.

Wightman, R. Pytorch image models. `https://github.com/rwightman/pytorch-image-models`, 2019.

Wu, C.-Y., Manmatha, R., Smola, A. J., and Krahenbuhl, P. Sampling matters in deep embedding learning. In *International Conference on Computer Vision*, 2017.

Wu, Z., Efros, A. A., and Yu, S. Improving generalization via scalable neighborhood component analysis. In *European Conference on Computer Vision*, 2018.

Zhai, A. and Wu, H.-Y. Classification is a strong baseline for deep metric learning. *arXiv preprint arXiv:1811.12649*, 2018.

Zhang, X., Yu, F. X., Kumar, S., and Chang, S.-F. Learning spread-out local feature descriptors. In *International Conference on Computer Vision*, 2017.

Zhao, K., Xu, J., and Cheng, M.-M. Regularface: Deep face recognition via exclusive regularization. In *Computer Vision and Pattern Recognition*, 2019.