Evaluating Contrastive Models for Instance-based Image Retrieval

Tarun Krishna, Kevin McGuinness and Noel O'Connor

tarun.krishna2@mail.dcu.ie

SFI Insight Centre for Data Analytics, Dublin City University, Ireland

ABSTRACT

In this work, we evaluate contrastive models for the task of image retrieval. We hypothesise that models that are learned to encode semantic similarity among instances via discriminative learning should perform well on the task of image retrieval, where relevance is defined in terms of instances of the same object. Through our extensive evaluation, we find that representations from models trained using contrastive methods perform on-par with (and outperforms) a pre-trained supervised baseline trained on the ImageNet labels in retrieval tasks under various configurations. This is remarkable given that the contrastive models require no explicit supervision. Thus, we conclude that these models can be used to bootstrap base models to build more robust image retrieval engines.

CCS CONCEPTS

• Computing methodologies → Visual content-based indexing and retrieval.

KEYWORDS

Deep learning, Contrastive learning, Self-supervised learning

ACM Reference Format:

Tarun Krishna, Kevin McGuinness and Noel O'Connor. 2021. Evaluating Contrastive Models for Instance-based Image Retrieval. In Proceedings of the 2021 International Conference on Multimedia Retrieval (ICMR '21), August 21–24, 2021, Taipei, Taiwan, ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3460426.3463585

1 INTRODUCTION

Large scale image retrieval, where the task is to search a large image collection for the most relevant image/content for a given query, is a fundamental task in computer vision. Since their inception, convolutional neural networks (ConvNets) [22, 40] have become the prominent approach for extracting descriptors for image retrieval. These descriptors perform very well in capturing the global semantics of an image and this has led to state-of-the-art results on many benchmark computer vision tasks [7, 18, 37].

The activations in the intermediate layers in ConvNets can be used as a descriptor for an image. These descriptors are often followed by some encoding methods for a compact representation. These encoding techniques range from traditional approaches of VLAD [21], BoW [25], and Fisher vectors [28], to simple pooling methods like Maximum Activation of Convolution (MAC) [2], Sum Pooling of Convolution (SPoC) [4], Regional-MAC [42], etc. The drawback of these methods is that these (off-the-shelf) network are trained to reduce inter-class variance through supervision on ImageNet classes, this might affect the performance of instance retrieval (i.e. retrieving images that represent the same object or scene as in a query), which is a more fine-grained task.

This drawback has been addressed in the literature by fine-tuning [13, 14, 35, 38]. We hypothesise that a simpler approach could be to retain a traditional off-the-shelf regime but instead use models that are trained based on instance-wise supervision using similarity-based learning. To this end, we investigate contrastive learning based methods, i.e. trained in an unsupervised fashion using contrastive loss [16, 44]. This learning regime relies on learning a meaningful embedding that captures inherent similarity between instances using discriminative approaches [44]. This work investigates the effectiveness of contrastive methods that capture this very idea of instance similarity. To summarize our contributions:

• we extensively evaluate contrastive methods as a fixed feature extractor across different benchmark;
• we provide experimental evidence showing that these models (trained without any explicit supervision) perform on par with a pre-trained supervised baseline (Table 1 and 2);
• we further investigate the role of the dimensionality of the feature embeddings for this task (Table 3).

2 RELATED WORK

Conventional image retrieval methods [26, 41] relied on bag-of-words features that exploit local invariant features such as SIFT [24] and large visual vocabularies (e.g. [31]). To aggregate local patches and build a global summary, encoding methods such as Fisher vectors [28] or VLAD [21], have also been proposed [15, 30, 34]. Since the introduction of Deep-ConvNets, [11, 18, 22, 40] there has been a paradigm shift to exploit deep features instead of hand-crafted ones. Intermediate layers in convolutional nets can be used as global or local descriptors. As a result, so-called off-the-shelf [3, 36] features can be used for retrieval. Based on this authors in [4] used sum pooling with a centre prior for aggregating features across spatial dimensions. Other conventional encoding techniques like VLAD [12] or Fisher kernels [29] have also been used in combination with these local feature maps. For example, in [25] proposed BoW-encodings of convolutional features for instance retrieval, whilst [42] proposed R-MAC using max activations over a grid of windows of different scales to obtain compact representations.

Most of the off-the-shelf features are trained on ImageNet [39] to reduce inter-class variance. However, this may degrade the performance of an instance-based retrieval system. One way to address this is to finetune the model as shown in [5, 13, 14, 35, 38]. In the
context of image retrieval, most of the finetuning has been performed on Landmark datasets [5], which further requires cleaning of non-related images and potentially expensive post-processing.

Another way is to exploit methods that are trained to reduce intra-class variance, as is the case in contrastive learning. Unlike in supervised learning, these approaches learn to discriminate among individual instances without any concept of categories. This work [44] discusses this notion of instance discrimination. Building on this, a simple formulation is presented in SimCLR [8]. The intuition behind these approaches is that they require large batch sizes on large GPU clusters. To address this drawback, the authors in [9, 17] introduced MoCo, which uses an online and momentum updated offline network that views contrastive learning as a dictionary lookup task. Intuitively, the ability to discriminate among individual instances inherently encoded through the learning makes contrastive learning a good candidate for the task of instance retrieval.

3 CONTRASTIVE MODELS

Contrastive learning refers to learning by comparison. This comparison is performed between positive pairs of “similar” and negative pairs of “dissimilar” inputs, which is achieved via a contrastive loss [8, 23] derived from Noise Contrastive Estimation (NCE) [16]. Minimising NCE is equivalent to maximizing mutual information (MI) as was formally shown in CPC [27] as InfoNCE. DIM [19] and AMDIM [6] further extend the idea of InfoNCE across multiple views and scales. One of the downsides of these approaches is that they require large batch sizes on large GPU clusters. To address this drawback, the authors in [9, 17] introduced MoCo, which uses an online and momentum updated offline network that views contrastive learning as a dictionary lookup task. Intuitively, the ability to discriminate among individual instances inherently encoded through the learning makes contrastive learning a good candidate for the task of instance retrieval.

4 EXPERIMENTS

This section describes our experimental setup for the evaluation.

4.1 Setup

We evaluate models on three standard benchmark datasets: Oxford5k [31], Paris6k [32], and INSTRE [43]. Retrieval performance is measured using mean Average Precision (mAP) following standard procedures for Oxford 5k and Paris 6k benchmarks and for INSTRE evaluating mAP over 1200 images as described in [20]. We further evaluate the performance on revised rOxford5k and rParis6k using the new evaluation protocol based on easy, medium, and hard ground truth labels [33]. For the revised benchmarks we report both mAP and mean precision@10,5 (mp@10, mp@5).

The goal is not to fine-tune the models but instead evaluate them as a fixed feature extractor to obtain visual descriptors. The base encoder of each of the models is some flavour of ResNet1 with varying complexity. To this end, we consider the output of the last convolutional layer, i.e. just before the adaptive pooling layer, as our descriptor, which leads to feature maps of size $\mathbb{R}^{C \times H \times W}$. To obtain compact representations we use R-MAC ($L = 3a$) [42] over spatial dimensions to get a fixed representation of size $\mathbb{R}^C$.

We further post-process the vectors by applying $L_2$ normalization, PCA-whitening, and $L_2$ normalization again.

We resize our input images to a fixed resolution of $724 \times 724$ giving a feature spatial dimension of $23 \times 23$ except in the case of AMDIM where the dimensions are $40 \times 40$. However, we downsample this to $23 \times 23$ to keep uniformity across the evaluation. Also, before running the final evaluation we first run each of the models in training mode ($\text{PyTorch} \ . \ \text{train()}$ just feed-forward) to tune the batch-normalization statistics to the current dataset and then finally test models in evaluation mode ($\text{model1.eval()}$).

Baseline. For comparing across all the contrastive models we use ResNet50 [18] trained on ImageNet as a fixed feature extractor as our pre-trained supervised baseline model. Note. For completeness we also evaluate a fine-tuned model [38], which uses Generalized Mean Pooling (instead of R-MAC) trained with Average Precision loss (GeM (AP))$^2$. The purpose of this is to provide an indicative upper bound to the evaluation scores.

Ranking. We consider global search (G) in this evaluation. We further integrate Global search with Average Query Expansion [10] (AQE), DataBase Augmentation (DBA), [1] and Diffusion [45] (DFS). For AQE we consider nearest neighbour $N = 10$, for DBA we consider $N = 20$ while combining both of these we consider $N = 1$ and $N' = 20$, based on the findings in [14].

We use a PCA dimension of 512 and evaluate on a global search for R-MAC representations unless otherwise stated.

4.2 Results

Table 1 compares different models along with different expansion techniques$^3$. For a naive global search on Oxford 5k, the contrastive approach achieves an mAP (%) of 59.40 while the baseline achieves 55.12. Overall best performance is achieved with AQE and DFS for the baseline model (82.64) but this mAP score for SimCLR$_{2x}$ (82.34) is in the same range as the former. We also include results

$^3$https://github.com/naver/deep-image-retrieval

$^3$here and in Table 2 GeM (AP) serves as a upper bound indicator rather than a benchmark.
of ensembling contrastive methods\(^4\), which seems to give a further performance boost (in red in Table 1). A similar inference could be drawn for Paris 6k where the best mAP for global is achieved by MoCo\(_{o2}\) (49.72) while the overall best (91.61) is achieved with AQE and DFS for SimClr\(_{2x}\). A further boost can be observed for the ensemble. In the case of the INSTRE dataset, we see a similar pattern for global search with mAP 44.76 corresponding to SimClr\(_{4x}\), while the overall best is achieved with AQE and DFS for the baseline (76.43) while the best result achieved for contrastive models is 75.02 for SimClr\(_{4x}\). This clearly indicates that contrastive methods trained to reduce intra class variance capture the notion of instance similarity which is being reflected in this evaluation. Also expansion techniques further boosts the performance over global search.

To further consolidate our findings, we also conducted an evaluation on the revised rOxford 5k and rParis 6k datasets as depicted in Table 2. On rOxford 5k SimClr\(_{2x}\) gives the best performance on all labels. mP@10 is almost 70% for the easy category, with the drop in

---

\(^4\)The R-MAC representations are concatenated and dimensionally reduced via PCA.
Table 2: Comparison on global ranking across different model. Bold (red) best performing ensemble if it exists.

| Dataset      | Method            | Easy mAP | mp@5 | mp@10 | Medium mAP | mp@5 | mp@10 | Hard mAP | mp@5 | mp@10 |
|--------------|-------------------|----------|------|-------|------------|------|-------|----------|------|-------|
|              |                   | mAP      | mp@5 | mp@10 | mAP        | mp@5 | mp@10 | mAP      | mp@5 | mp@10 |
|              | Baseline          | 45.65    | 67.55| 61.81 | 32.89      | 64.00| 58.57 | 12.10    | 23.71| 19.43 |
|              | SimCir1x          | 47.17    | 69.78| 63.90 | 31.98      | 65.24| 57.52 | 9.52     | 20.57| 15.86 |
|              | SimCir2x          | 54.95    | 76.69| 69.49 | 38.54      | 73.14| 65.33 | 14.11    | 31.21| 22.93 |
|              | SimCir4x          | 54.65    | 74.80| 68.37 | 38.57      | 73.71| 65.05 | 14.05    | 30.07| 22.79 |
|              | MoCo1b            | 48.29    | 68.48| 64.36 | 33.57      | 64.43| 58.58 | 9.27     | 20.05| 15.62 |
| rOxford 5k   | MoCo2q            | 52.69    | 72.33| 65.86 | 36.49      | 67.24| 59.24 | 10.72    | 23.38| 18.67 |
|              | AmDim             | 21.24    | 38.31| 31.35 | 17.54      | 36.38| 32.57 | 4.11     | 5.92 | 6.33  |
|              | SimCir2x+SimCir1x | 55.23    | 76.15| 68.87 | 39.60      | 74.29| 65.64 | 15.01    | 29.43| 23.97 |
|              | SimCir2x+AmDim   | 52.08    | 72.28| 65.51 | 36.19      | 66.86| 58.57 | 10.20    | 21.64| 17.64 |
|              | SimCir2x+MoCo2q  | 55.11    | 76.47| 69.35 | 38.46      | 70.52| 62.57 | 12.51    | 27.12| 21.83 |
|              | SimCir4x+MoCo2q  | 53.98    | 72.40| 67.90 | 38.31      | 70.10| 64.69 | 13.38    | 28.43| 23.13 |
|              | SimCir4x+AmDim   | 49.23    | 70.76| 64.00 | 35.72      | 67.62| 61.33 | 11.72    | 23.79| 18.80 |
|              | GeM (AP)          | 64.07    | 84.93| 80.56 | 51.03      | 89.43| 83.86 | 30.30    | 54.86| 44.00 |

Table 3: Comparison of mAP (%) across different PCA dimension and the true dimension.

| Dataset      | Method            | Easy PCA-Whitening | Medium PCA-Whitening | Hard PCA-Whitening |
|--------------|-------------------|--------------------|----------------------|-------------------|
|              |                   | 32 | 64 | 128 | 256 | 512 | 1024 | 2048 | True dim. |
|              | Baseline          | 48.31 | 54.61 | 56.68 | 57.52 | 55.12 | 49.44 | 37.23 | 58.47 |
|              | SimCir1x          | 34.96 | 44.96 | 54.17 | 54.79 | 51.12 | 44.16 | 34.94 | 50.63 |
|              | SimCir2x          | 36.79 | 47.96 | 60.32 | 61.68 | 58.59 | 49.82 | 24.62 | 53.72 |
|              | SimCir4x          | 33.52 | 43.86 | 57.77 | 61.70 | 59.40 | 53.83 | 44.71 | 41.94 |
|              | MoCo1b            | 29.90 | 41.95 | 54.62 | 57.38 | 56.76 | 50.20 | 39.37 | 38.73 |
| rOxford 5k   | MoCo2q            | 39.92 | 47.88 | 57.92 | 60.43 | 58.36 | 52.14 | 40.05 | 51.76 |
|              | AmDim             | 11.52 | 13.30 | 21.71 | 28.71 | 36.95 | 37.58 | 38.73 | 16.10 |
|              | Baseline          | 72.22 | 71.83 | 63.65 | 53.40 | 41.46 | 28.99 | 17.89 | 68.36 |
|              | SimCir1x          | 69.90 | 73.29 | 66.62 | 54.58 | 42.28 | 30.70 | 20.06 | 66.60 |
|              | SimCir2x          | 73.21 | 77.90 | 69.25 | 57.83 | 44.48 | 33.05 | 23.27 | 72.20 |
| Paris 6k     | SimCir4x          | 77.16 | 78.04 | 69.19 | 57.29 | 44.96 | 35.02 | 25.79 | 72.89 |
|              | MoCo1b            | 56.84 | 63.63 | 61.87 | 54.07 | 43.22 | 32.57 | 21.63 | 53.66 |
|              | MoCo2q            | 70.08 | 71.15 | 71.24 | 61.66 | 49.72 | 35.73 | 22.98 | 69.99 |
|              | AmDim             | 21.49 | 33.96 | 41.19 | 40.85 | 34.87 | 26.82 | 17.85 | 25.00 |
|              | Baseline          | 26.44 | 34.92 | 38.68 | 35.35 | 36.64 | 29.25 | 20.20 | 33.03 |
|              | SimCir1x          | 16.47 | 23.27 | 29.01 | 30.35 | 27.51 | 21.85 | 19.57 | 21.85 |
|              | SimCir2x          | 18.42 | 26.83 | 35.88 | 39.67 | 36.68 | 29.64 | 21.57 | 25.94 |
|              | SimCir4x          | 18.92 | 28.97 | 40.45 | 46.99 | 44.76 | 36.68 | 27.65 | 28.98 |
|              | MoCo1b            | 20.08 | 27.85 | 33.77 | 36.24 | 33.44 | 26.66 | 18.46 | 23.01 |
|              | MoCo2q            | 19.00 | 27.38 | 34.34 | 36.22 | 33.36 | 26.86 | 18.79 | 26.22 |
|              | AmDim             | 10.65 | 15.88 | 20.55 | 24.15 | 24.34 | 20.45 | 14.45 | 10.25 |

Effect of descriptor dimension on performance. Table 3 reports our findings for global search\(^3\). Interestingly, true dimensions ($L_2$ normalized R-MAC representations) appear to perform worse for almost all the models. The best dimension varies across the dataset but it is never the true dimension. This could be attributed to dimensions with small principal components being noisy and redundant and adversely affecting performance.

5 CONCLUSION

This work evaluated contrastive models for the task of instance-based image retrieval. Our evaluation found that these methods are on par with those trained on class labels. In fact, in many settings in Table 1, 2 contrastive approaches surpass the supervised model. The quantitative evaluation shows that these contrastive methods can easily surpass supervised models without any explicit supervision.

6 ACKNOWLEDGMENT

This work has emanated from research supported by Science Foundation Ireland (SFI) under Grant Number SFI/12/RC/2289_P2, co-funded by the European Regional Development Fund and Xperi FotoNation.

---

\(^3\)Comparison across the horizontal dimension (columns)
