Relational Self-supervised Distillation with Compact Descriptors for Image Copy Detection

Juntae Kim Sogang University Seoul, South Korea

jtkim1211@sogang.ac.kr

Sungwon Woo Sogang University Seoul, South Korea

swwoo@sogang.ac.kr

Jongho Nang Sogang University Seoul, South Korea

jhnang@sogang.ac.kr

Abstract

Image copy detection is a task of detecting edited copies from any image within a reference database. While previous approaches have shown remarkable progress, the large size of their networks and descriptors remains disadvantage, complicating their practical application. In this paper, we propose a novel method that achieves a competitive performance by using a lightweight network and compact descriptors. By utilizing relational self-supervised distillation to transfer knowledge from a large network to a small network, we enable the training of lightweight networks with a small descriptor size. We introduce relational self-supervised distillation for flexible representation in a smaller feature space and applies contrastive learning with a hard negative loss to prevent dimensional collapse. For the DISC2021 benchmark, ResNet-50/EfficientNet-B0 are used as a teacher and student respectively, the micro average precision improved by 5.0%/4.9%/5.9% for 64/128/256 descriptor sizes compared to the baseline method.

1. Introduction

Image copy detection (ICD) is a task of detecting edited copies from an existing database, where each image instance is treated as an individual category such as instance matching problem [8]. This is widely used in online sharing platforms, including social networks, to filter content for copyright protection. Image copy detection has been regarded as a sub-task of instance-level recognition over the past years [9]. Their approaches employ global descriptors [2, 24, 32, 33] and local descriptors [25, 37, 42], which focus on identifying images belonging to the same particular object. However, ICD addresses a more nuanced challenge by focusing on exact image copies that often involve severe transformations such as re-encoding, resizing, merging, cropping, warping, or color distortion. To better address these challenges, recent advancements have leveraged

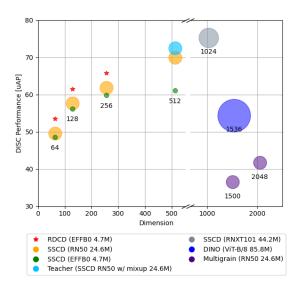


Figure 1. Comparison of RDCD(Ours) and other image copy detection methods. RDCD utilizes a lightweight network and achieves high performance despite its compact descriptor sizes.

self-supervised learning (SSL) methods such as [5,6,15,49] to directly apply data augmentation as a training objective [31,45].

ICD involves two key challenges. The first is the large-scale of detecting system. In large-scale systems, millions of images in the database are pre-processed offline into descriptors and stored. Online query images are then converted into descriptors in real-time, and a nearest-neighbor search method is employed, similar to image search systems [14,19,22]. Ideally, creating smaller descriptor sizes is an effective method to reduce search time and storage space for descriptors. In general, lightweight architectures have been introduced to convert images into descriptors at faster speeds and with smaller descriptor sizes. However, several studies [13, 36] have found that lightweight networks with limited representation power cannot directly employ SSL to produce acceptable performance. Previous studies

have introduced SimCLR to train ResNet-50 (RN-50) and ResNeXt-101, and we find that directly applying SimCLR to lightweight networks results in poor performance. Another method uses principal component analysis (PCA) to create small-sized descriptors, but it leads to significant performance degradation [31]. A second challenge for ICD stems from the difficulty of distinguishing hard negative samples. In ICD, there are numerous hard negative samples that are visually similar to the original image but are not edited copies. These include images taken from different camera angles or images of the same location captured at different times, which can significantly complicate the identification process. In ICD, each individual instance is treated as a separate category, making it crucial for the embedding space to achieve a uniform distribution. This arrangement helps to further each instance from its nearest neighbor (hard negative). In [31], the Kozachenko-Leononenko estimator [35] is an entropy regularizer used to promote a uniform embedding distribution, thus ensuring that the distances from embedding regions are more comparable. By achieving a uniform distribution, the approach makes full use of all dimensions, thereby addressing the issue of dimensional collapse. Through experiments, we confirmed that training a student model with a teacher model previously trained using the entropy regularizer partially alleviated the issue of dimensional collapse; in other words, the student model still does not fully utilize all dimensions of the descriptors, continuing to struggle with dimensional collapse.

In this work, we introduce Relational Self-supervised Distillation with Compact Descriptors (RDCD), which utilizes both relation-based self-supervised distillation (RSD) and the HN loss. RSD is instrumental in accurately capturing the intricate relationships between teacher model descriptors, to ensure their effective representation in a reduced feature space. Existing SSL with knowledge distillation (SSL-KD) methods use feature-based knowledge distillation (FKD) [12] or RSD [1, 10] for pre-training in classification tasks. FKD transfers knowledge by aligning embeddings of the student model with those of the teacher model using mean squared error loss. However, we found that FKD fails to prevent dimensional collapse, leading to performance degradation in ICD. We create a separate instance queue for the teacher to make the student mimic the pairwise similarity generated by the teacher through Kullback-Leibler (KL) divergence, allowing the student network to generate discriminative descriptors in its own feature space.

Although our distillation method is similar to previous SSL-KD [1, 10], it ultimately differs in the type of information it aims to distill. The main purpose of these methods is to provide advantages for downstream tasks as pre-training methods in lightweight networks. Their objective is to enrich the intermediate representations with

more transformation-covariant features, embedding useful information such as color and orientation [6]. In contrast, our objective is to learn transformation-invariant representations at the final layer rather than obtaining transformation-covariant intermediate features. This design is crucial as the ICD task requires representations that go beyond instance discrimination to a tighter level, which is distinctly different from instance retrieval or classification tasks. In the ICD task, transformation-invariant features are essential, while intermediate representations with transformation-covariant characteristics degrade performance, as illustrated in [31].

We conduct comparative experiments with SimCLR [6], MoCo-v2 [7], FKD, and RSD to analyze the effectiveness of each method in training efficient lightweight networks for ICD. We test our approach on various teacher and student networks, including convolutional neural networks and vision transformers, to demonstrate its architecture-agnostic characteristics. The main contributions of this work are summarized as follows:

- We introduce RDCD, a novel approach that combines RSD with HN loss to train lightweight networks for ICD in a self-supervised manner. Our method leverages the relational information between descriptors generated by a teacher network to guide the learning of a student network with a compact descriptor size.
- We employ HN loss to prevent the dimensional collapse that can occur in SSL with compact descriptors, ensuring a structured and informative feature space.
- Through extensive experiments, we validate the effectiveness of our RDCD approach with various architectures, including CNNs and ViTs.

2. Related Work

2.1. Existing Image Copy Detection Methods

The ICD task has seen significant advancements since the 2021 Image Similarity Challenge [8] hosted by Meta AI Research, which spurred numerous research developments. Previous approaches utilize either contrastive selfsupervised learning or deep metric learning. [47] use contrastive loss and cross-batch memory. [27] employ Arcface loss based metric learning and use a drip training technique to incrementally increase the number of classes during training. [23] use deep metric learning through hard sample mining with triplet loss and classification loss. In subsequent years, [31] apply InfoNCE loss with entropy regularization. Different from previous winning solutions which use numerous techniques to enhance performance, such as ensembles and post-processing, our approach follows the same evaluation dataset and basic evaluation methods from [31]. Additionally, we trained our model using a purely unlabeled dataset, without employing any ground truth labels. [11] applies imperceptible modifications to images called "activation", when using an ICD model in conjunction with Approximate Nearest Neighborhood indexing, ensuring the transformed image is positioned at the center of the quantized cluster. To address the efficiency, our method aims to directly reduce descriptor sizes without relying on additional indexing techniques [20, 38].

2.2. Knowledge Distillation

Using a compact model to approximate a function learned by a more comprehensive, higher-performing model was first introduced in [4]. This concept was expanded in [16], where the student model was trained to mimic the teacher's softened logits, a process referred to as knowledge distillation. Feature distillation has been garnering significant attention recently. [34] added a new dimension by proposing a hint-based training scheme for the alignment of feature maps, known as feature distillation, and chose the L2 distance as the metric for comparing the two feature maps. Similarly, [48] suggested the transfer of spatial attention maps from a high-performing teacher network to a smaller student network. [46] devised the idea of transferring 'flow'—defined as the inner product between features from two layers—to the student rather than knowledge. [29] directly aligned the probability distributions of the data between the teacher's and student's feature spaces. The success of these knowledge distillation methods can primarily be attributed to the insightful knowledge embedded in the logits of the teacher model.

2.3. Relation-based Knowledge Distillation

While conventional knowledge distillation only extracts the information for a single data point from the teacher, similarity-based distillation methods [1,10,28,29,39,41,43] learn the knowledge from the teacher in terms of similarities between data points. A pairwise similarity matrix has been proposed to retain the interrelationships of similar samples between the representation space of the teacher and student models. CompRess [1] and SEED [10] which are approaches related to our method, use similarity-based distillation to compress a large self-supervised model into a smaller one. However, our method differs in that we distill the relationship at the final layer to learn the transformation-invariant descriptor itself, whereas previous approaches enrich the intermediate representations with more transformation-covariant features.

3. Methodology

3.1. Overall Architecture

The overall architecture of the proposed RDCD approach is presented in Fig 2. In RDCD, the student network

is trained using three different objectives. Because the student network $f^S(\cdot)$ is the encoder that we want to improve, we freeze teacher network $f^T(\cdot)$ which is a pre-trained encoder by training using an off-the-shelf method.

Given an image x, the teacher network extracts its representation h^T , in which $h^T \in \mathbb{R}^{D_T}$ is the final feature without any classifier. The student network extracts two representations h^S and $h^{S'}$ from input images augmented in different ways. An FC layer is employed immediately after the student network to match the dimensionalities of h^T and h^S . Furthermore, we also add an additional projector for contrastive learning by the student. This projector decreases the dimensions of representation and is used to calculate pairwise similarities for the computation of the contrastive loss function. We use SimCLR and MoCo-v2 as our contrastive learning methods, but any contrastive-based learning method can be adopted.

3.2. Relational Self-supervised Distillation

Relational Self-supervised Distillation (RSD) is a method that transfers instance relations from a teacher network to a student network. This process involves the creation of an instance queue within the teacher's network, designed to store the teacher's instance embeddings. This queue serves as a reference point for the student network, facilitating the transfer of knowledge. When a new sample is introduced, its similarity scores are calculated against all instances in the queue (q_k) using both the teacher and student networks. The similarity scores are typically computed using cosine similarity.

The key objective is to align the similarity score distribution generated by the student network with that of the teacher network. This alignment is achieved by minimizing the KL-divergence between the similarity score distributions of the two networks, ensuring a close match in their respective interpretations of the instance similarities. We apply the same augmentations into an image x_i as a batch and map the embeddings $h^T = f_{\theta}^T(\tilde{x_i})$ and $h^S = f_{\theta}^S(\tilde{x_i})$ where $h^T, h^S \in \mathbb{R}^D$ and f_{θ}^T and f_{θ}^S denote the teacher and student network, respectively.

We compute the cosine similarity between l2-normalized descriptors from the teacher network and the queue. For the teacher and student, the similarities are:

$$sim(h_i^T, q_j^T) = \left[h_i^T / \|h_i^T\|_2\right] \cdot \left[q_j^T\right]^\top \tag{1}$$

$$sim(h_i^S, q_j^T) = [h_i^S / ||h_i^S||_2] \cdot [q_j^T]^\top$$
 (2)

where sim represents the cosine similarity, and q_j^\top denotes the transposed j-th component in the teacher's queue Q^T to compute cosine similarity. We calculate the probability of the i-th instance with the j-th component in the teacher's queue.

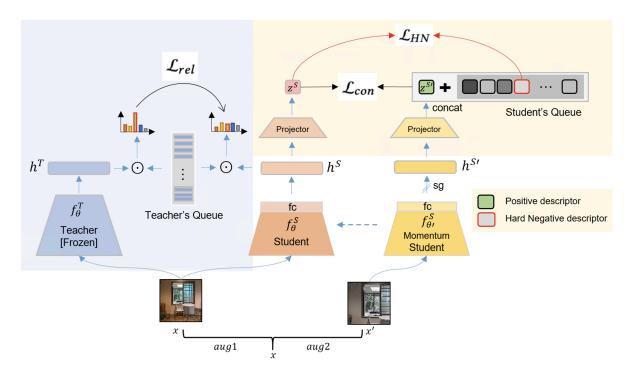


Figure 2. Overall pipeline of proposed Relational Self-supervised Distillation for Image Copy Detection (RDCD).

$$p_{i,j}^{T} = \frac{exp(sim(h_i^{T}, q_j^{T})/\tau^{T})}{\sum_{q^{T} \sim Q} exp(sim(h_i^{T}, q^{T})/\tau^{T})}, \ j \in [1 \dots K] \quad (3)$$

$$p_{i,j}^{S} = \frac{exp(sim(h_{i}^{S}, q_{j}^{T})/\tau^{S})}{\sum_{q^{T} \sim O} exp(sim(h_{i}^{S}, q^{T})/\tau^{S})}, \ j \in [1 \dots K] \quad (4)$$

where τ^T and τ^S are the temperature parameters of the teacher and student networks, respectively. K is the size of the teacher's queue. Also, $p_{i,j}^T$ denotes the similarity score between the embeddings h_i^T from the teacher network and the embeddings in the teacher's queue. In the same way, $p_{i,j}^S$ denotes the similarity score between the embeddings h_i^S from the student network and the embeddings in the teacher's queue. The objective of our RDCD is to minimize the KL-divergence between the probabilities over all input instances for the teacher and student networks. The final loss function is determined as follows:

$$\mathcal{L}_{rel} = \sum_{i}^{N} KL(p_i^T \mid\mid p_i^S)$$

$$= \arg\min_{\theta_S} \sum_{i}^{N} \sum_{j}^{K} -\frac{\exp(\operatorname{sim}(h_i^t, q_j^T)/\tau^T)}{\sum_{q^T \sim Q} \exp(\operatorname{sim}(h_i^t, q^T)/\tau^T)} \cdot \log\left(\frac{\exp(\operatorname{sim}(h_i^s, q_j^T)/\tau^S)}{\sum_{q^T \sim Q^T} \exp(\operatorname{sim}(h_i^s, q^T)/\tau^S)}\right)$$
(5)

3.3. Contrastive Learning for ICD

The size of the embeddings within the architecture of lightweight networks is generally small, and it is crucial to maintain this size during training while benefiting from knowledge distillation. To ensure this, we attach a linear projector to the end of the student network that is designed to reduce the representation size, resulting in smaller embeddings. We use these final embeddings as our descriptors, which allows for simultaneous training using our relational distillation approach. We follow the MoCo-v2 [7] training procedure by utilizing a negative instance queue generated by the momentum encoder. We employ the InfoNCE [26] loss to guide contrastive learning, which is calculated as follows:

$$\mathcal{L}_{con} = -\frac{1}{N} \sum_{i=1}^{N} \log \frac{\exp(\sin(z_i^S, z_i^{S'})/\tau)}{\sum_{q^S \sim Q^S} \exp(\sin(z_i^S, q^S)/\tau)}$$
(6)

For each pair index i, z_i^S , and $z_i^{S'}$ are the representations of the two augmented views of the same image (i.e., a positive pair). $\sin(z_i^S, z_i^{S'})$ is a function measuring the similarity between these descriptors, typically a dot product. τ is a temperature parameter that scales the similarity scores. The denominator sums the exponentiated similarities of z_i^S with all negative descriptors q^S in all students' queue Q^S .

3.4. The Hard Negative Loss

In the copy detection scenario, the primary challenges for training arise from the hard negative pairs and the fact that the use of small descriptors renders the environment susceptible to dimensional collapse. To address this, we employ the Hard Negative (HN) loss [21] to finely control the entropy and minimize the hard negative pairs. We also introduce an additional term that ensures that a hard negative pair can push their descriptors apart.

$$\mathcal{L}_{hn} = -\frac{1}{N} \log \sum_{i=1}^{N} \max_{j \in n_{i,j}} (1 - S_{i,j})$$
 (7)

 $n_{i,j}$ denotes the row indices of the negative pairs for the i-th row of the similarity matrix.

The objective function of the student network is a combination of the contrastive loss (con), the Relational Self-supervised Distillation (rel) loss, and the hard negative (HN) loss. The final objective can be expressed as:

$$\mathcal{L}_{RDCD} = \lambda_{rel} \mathcal{L}_{rel} + \lambda_{con} \mathcal{L}_{con} + \lambda_{hn} \mathcal{L}_{hn}$$
 (8)

4. Experiments

4.1. Dataset

DISC2021 is a dataset consisting of training, reference and query images, that is used for the Image Similarity Challenge [9]. The query image set contains strong autoaugmented and human-augmented images, and distractive images that may not appear in the reference set. The training set is designed to train a model without any labels, and the reference set is used for searching the background as a database.

Copydays [17] is a dataset designed to detect copied content, that consists of 157 original images and 2,000 queries with cropping, jpeg compressing and strong augmentations. We add 10k distractors from YFCC100M [40], a common practice [3, 5] that is known as CD10K. We evaluate our method based on the mean average precision(mAP) and micro average precision(μAP) for the strongly transformed copies.

NDEC [44] dataset incorporates HN distractors, making it more advanced than conventional ICD datasets. The basic data is derived from DISC2021 while the HN images are sourced from OpenImage. Similar to DISC2021, the NDEC dataset is organized into training, query, and reference sets. The training data includes 900,000 basic images and adds HN pairs (100,000 x 2 paired images) to enhance the difficulty level of ICD. The test data contains 49,252 query images and 1,000,000 reference images, with 24,252 of the query images being HN.

4.2. Evaluation Metrics

To enable a fair comparison with state-of-the-art methods [5,31], we use μAP [30], accuracy at 1, and recall at precision 90 as the evaluation metrics. The equation for μAP is as follows:

$$\mu AP = \sum_{i=1}^{N} P(i) \triangle r(i) \in [0, 1]$$
 (9)

where P(i) is the precision at position i of the sorted list of pairs, $\triangle r(i)$ is the difference in the recall between position i and i-1, and N is the total number of returned pairs for all queries. Any detected pair for a distractor query will decrease the average precision, thus all queries are evaluated together by merging the returned pairs for all queries, sorting them by confidence, and generating a single precisionrecall curve. This is different from mAP, also known as macro-AP [30], where the average precision is computed separately per query and then averaged over all queries. μAP considers confidence values to capture all queries, while mAP only considers the query in the ground truth. Thus, μAP is a more accurate metric in our context because our queries contain the images not contained in the references. The former is more appropriate for ICD, while the latter is typically used in retrieval tasks.

Table 1. Results for the DISC2021 showing different methods, network architectures (ResNet-50, ViT-B/8, ViT-B/16, ResNeXt-101, EfficientNet-B0), model sizes, and performance metrics (micro Average Precision μ AP and micro Average Precision with score normalization μ APSN). * means our implementation.

Method	Network	Size	μ AP	μ APsn
Multigrain	RN-50	1500	16.5	36.5
Multigrain	RN-50	2048	20.5	41.7
DINO	ViT-B/8	1536	32.6	54.4
DINO	ViT-B/16	1536	32.2	53.8
SSCD	RN-50	512	61.5	72.5
SSCD	ResNeXt-101	1024	63.7	75.3
using lightweight*				
SSCD	EN-B0	64	38.2	48.5
SSCD	EN-B0	128	43.5	56.2
SSCD	EN-B0	256	46.0	59.8
SSCD	EN-B0	512	43.6	61.1
ours				
RDCD	EN-B0	64	43.9	53.5
RDCD	EN-B0	128	50.0	61.1
RDCD	EN-B0	256	52.7	65.7

4.3. Training Implementation

We train all networks using the pretraining parameters from ImageNet-1K. Batch normalization statistics are synchronized across all GPUs. We adhere to the hyperparameters outlined in SimCLR [6], MoCo-v2 [7] and SSCD [31]. The batch size is set at N=256, with a resolution of 224 x 224 for training and 288 x 288 for inference. We use a learning rate of 1.0 and a weight decay of 10^{-6} . We train all models for 100 epochs. We employ a cosine learning rate scheduler with a warmup period of five epochs and use the Adam optimizer. For our knowledge distillation process, we set the temperature at 0.04 for the teacher and 0.07 for the student. All experiments are conducted on an NVIDIA A100 80GB GPU. To ensure reproducibility, we maintain consistent seed values throughout all of the experiments.

4.4. Results

Table 1 presents the results for the DISC2021 dataset. The first section of the table summarizes the performance of different methods with varying descriptor sizes, while the second section describes the performance of our baseline SSCD [31] strategy using a lightweight network with descriptor sizes of 64, 128, and 256. When training using a lightweight network, we do not use a mix-up for the fair comparison of our methods. The third section shows the results of the proposed RDCD approach.

When compared with the methods outlined in the first section, our RDCD method achieves competitive performance despite using significantly smaller descriptors and a more compact network size. It is noteworthy that the RDCD method, with a descriptor size of 64, achieves a μAP_{SN} of 53.5, which is comparable to the DINO method that utilizes a ViT-B/16 network with a descriptor size of 1536, yielding a μAP_{SN} of 53.8. This comparison is significant, as the descriptor size of RDCD is 24 times smaller.

When evaluated against an equivalent lightweight architecture, RDCD consistently outperforms SSCD across all three dimensions assessed. We employ the SSCD model RN-50, which has a μ APsN of 72.5, as the teacher network to provide knowledge to the lightweight network. When we apply our methods to EfficientNet-B0, RDCD outperforms SSCD. Remarkably, RDCD with a descriptor size of 128 achieves a μ APsN of 61.1, which is competitive and matches the performance of SSCD with a descriptor size of 512. This indicates that our method is capable of delivering strong results even with smaller descriptors.

RDCD also outperforms SSCD in terms of mAP using the CD10K dataset (Table 2). Specifically, when using EfficientNet-B0 with a descriptor size of 128, RDCD achieves an mAP of 79.2, which is significantly higher than the best result for SSCD. With a descriptor size of 256, the performance improvement is even more pronounced, with RDCD reaching a mAP of 81.4.

For the NDEC dataset, Table 3 shows that our method shows competitive performance compared to the teacher network SSCD-RN50, which achieves a μ AP_{SN} of 23.3 with

Table 2. Results on the CD10K (Copydays+ 10k distractors). We compare different methods, network architectures (ResNet-50, ViT-B/8, ViT-B/16, ResNeXt-101, EfficientNet-B0), model sizes, and performance metrics (mean Average Precision (mAP) and micro Average Precision (μ AP). * means our implementation.

Method	Network	Size	mAP	μ AP
Multigrain	RN-50	1500	82.3	77.3
DINO	ViT-B/8	1536	85.3	91.7
DINO	ViT-B/16	1536	80.7	88.7
SSCD	RN-50	512	85.0	97.9
SSCD	ResNext-101	1024	91.9	96.5
using lightweight*				
SSCD	EN-B0	64	64.1	95.6
SSCD	EN-B0	128	71.9	97.0
SSCD	EN-B0	256	76.6	97.4
SSCD	EN-B0	512	77.4	97.4
ours				
RDCD	EN-B0	64	72.4	94.5
RDCD	EN-B0	128	79.2	96.1
RDCD	EN-B0	256	81.4	95.4

a descriptor size of 512. Our proposed RDCD approach, employing EN-B0 network with descriptor sizes of 64, 128, and 256 achieves a μ AP_{SN} of 17.3, 19.5, and 21.3, respectively. This demonstrates that across all compact descriptor sizes, RDCD surpasses the performance of the previous SSCD when using the same lightweight architecture.

Table 3. Comparison of NDEC results. * means our implementation.

Method	Network	Size	μAP	$\mu \mathrm{AP}_{\scriptscriptstyle\mathrm{SN}}$
DINO	ViT-B/8	1536	16.2	22.8
DINO	ViT-B/16	1536	18.1	26.2
SSCD	RN-50	512	42.4	46.6
using lightweight*				
SSCD	EN-B0	64	28.9	32.8
SSCD	EN-B0	128	33.0	37.5
SSCD	EN-B0	256	35.0	40.1
SSCD	EN-B0	512	34.5	40.8
ours				
RDCD	EN-B0	64	31.4	34.7
RDCD	EN-B0	128	34.9	38.9
RDCD	EN-B0	256	37.5	42.5

5. Discussion

5.1. Effect of RSD with HN Loss

Our method employs an additional projector to reduce the descriptor size, which creates a network with a twolayer MLP. This configuration can lead to implicit regu-

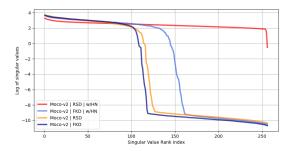


Figure 3. Log of singular values for a descriptor size of 256, with and without HN loss.

larization, due to the interaction between the weight matrices across different layers. Implicit regularization typically restricts the ability of a network to learn diverse features, leading to dimensional collapse with contrastive SSL [18]. However, when applying HN loss together with RSD, we show that dimensional collapse does not occur even with the use of an MLP, thus reducing the impact of implicit regularization.

To verify this, we calculate the singular values of the final descriptor of a model with a descriptor size of 256 under four conditions: with the use of RSD, FKD and with and without the use of HN loss. We compute the descriptors for 50k queries from the DISC21 dataset, calculate their covariance, and perform singular value decomposition on the covariance matrix. Subsequently, we extract the singular values and apply a logarithmic operation to them. As illustrated in Figure 3, RDCD has a full rank, while the absence of HN loss results in dimensional collapse. It can also be observed that applying HN loss with FKD does not have a significant effect.

We further visualize the difference between positive similarity and hard negative similarity across multiple dimensions experimented in MoCo-v2 — RSD environment, comparing scenarios with and without the application of HN loss. We analyze a sample of 5,000 distinct queries extracted from the DISC21 dataset, which are separate from the queries used in the standard evaluation process for generality, to calculate and assess the differences in the similarity between positive and hard negative samples. As Figure 4 shows, across all descriptor sizes, the application of HN loss resulted in a more pronounced disparity compared to when it was omitted. This observation implies that HN loss effectively enhances the separation between positive and hard negative samples. Furthermore, we noted that as the dimensionality increases, the difference between positive and hard negative similarity also enlarges. We conjecture that this trend may contribute to improved performance in higherdimensional descriptors.

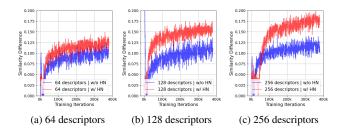


Figure 4. Comparison of the difference in similarity between positives and nearest negatives with and without the use of HN Loss.

Table 4. Comparison of other distillation. All methods are trained with EfficientNet-B0 network. We do not use Hard Negative loss for SSCD method as they use Koleo regularizer.

Hard Negative Loss			I	No	Yes	
Method	RSD	FKD	μAP	$\mu \mathrm{AP}_{\scriptscriptstyle\mathrm{SN}}$	μAP	μ APs
SSCD			43.5	56.2	-	-
SimCLR		\checkmark	46.2	58.7	48.4	59.9
	\checkmark		47.4	59.6	49.6	61.4
	\checkmark	\checkmark	47.5	59.6	50.8	61.5
MoCo-v2		\checkmark	40.3	57.5	44.8	59.1
	\checkmark		47.1	60.7	50.0	<u>61.1</u>
	\checkmark	\checkmark	46.2	59.3	49.6	61.0
256 descriptor	rs					
Hard Negative Loss			No		Yes	
Method	RSD	FKD	μAP	$\mu \mathrm{AP}_{\scriptscriptstyle\mathrm{SN}}$	μAP	μ APs
SSCD			46.0	59.8	-	-
SimCLR		\checkmark	51.7	64.0	51.4	64.5
	\checkmark		52.2	65.6	52.7	65.8
	\checkmark	\checkmark	52.4	65.5	49.9	64.6
		✓	41.3	56.6	46.1	60.1
MoCo-v2		•				
MoCo-v2	✓	•	47.2	61.0	52.9	65.7

5.2. Comparison with Other Forms of Distillation

To further demonstrate the efficacy of RSD, we conduct a comparative analysis with FKD by 1) employing FKD alone, 2) employing RSD alone, and 3) integrating both FKD and RSD (Table 4). We also employ SimCLR-style contrastive learning. When FKD and RSD are combined, the best performance is observed in several cases, suggesting that combining both distillation methods can further improve performance. Nevertheless, RSD consistently outperforms FKD in all cases, with RSD alone achieving performance on par with the combined use of both distillation methods. This indicates that RSD is sufficiently robust, and FKD does not significantly alter the outcome. It confirms that RSD can operate alone to enhance the performance of compact descriptors. Additional experiments with other distillation methods are presented in Appendix A.

5.3. Ablation Study

Effectiveness of Each Loss Term. To further assess the effectiveness of RDCD, we investigate the impact of different loss components on its performance, specifically, RSD loss and HN loss as shown in Table 5. In this section, we present an additional metric, the rank preserving ratio (RPR), which is the rank divided by the descriptor dimension. A low RPR indicates dimensional collapse, and RPR value of 1 indicates that the descriptor makes full use of all dimensions. All models are trained on the DISC21 dataset with 100 epochs and SSCD-RN50-w/mixup is employed as the teacher model. Our result shows that with the addition of RSD loss and HN loss, the performance improves substantially, which indicates that RSD and HN loss can benefit the performance of RDCD.

Table 5. Ablation study on the loss terms in RDCD. Descriptor sizes of 64, 128, and 256 were examined, and a consistent MLP configuration was maintained across all experiments for a fair comparison.

	Loss		MLP-d	A D	A D
\mathcal{L}_{con}	\mathcal{L}_{rel}	\mathcal{L}_{hn}	MILP-u	μ AP	$\mu \text{AP}_{\text{SN}}$
MoCo-v2					
\checkmark			1280/512/64	10.9	14.4
\checkmark			1280/512/128	11.9	17.2
\checkmark			1280/512/256	16.5	26.4
Effectiven	ess of RSD loss				
\checkmark	\checkmark		1280/512/64	39.9	52.1
\checkmark	\checkmark		1280/512/128	47.1	60.7
\checkmark	\checkmark		1280/512/256	47.2	61.0
Effectiven	ess of HN loss				
\checkmark	\checkmark	\checkmark	1280/512/64	43.9	53.5
\checkmark	\checkmark	\checkmark	1280/512/128	50.0	61.1
✓	✓	✓	1280/512/256	52.9	65.7

Influence of λ_{rel} and λ_{hn} . λ_{rel} and λ_{hn} are the weights of RSD loss and HN loss, respectively. First, we analyze the influence of contrastive loss and RSD loss (Table 6). As shown in the third row, a contrastive loss of 1 and RSD loss of 10 achieves the best performance, which indicates that knowledge distillation is essential for the model with better performance. The HN loss is also varied with the contrastive loss and RSD loss fixed at 1 and 10, respectively, showing that the best performance occurs when the HN loss is 5. However, for larger values, the performance drops to 0, (i.e. collapses) because the network focuses more on HN loss rather than contrastive and distillation loss. We further calculate RPR of each experiment. In the first section, where only contrastive loss and RSD loss are employed, we observe that RPR gradually increases incrementally with an increase in λ_{rel} . This suggests that the student model partially alleviates the dimensional collapse which is guided by a teacher model that has been trained with an entropy regularizer (SSCD). Subsequently, in the second section, where HN loss is incorporated, the student model finally utilizes the full dimensions of the descriptors.

Table 6. Ablation study to evaluate the impact of the loss ratio. All experiments are performed with a descriptor size of 256.

λ_{con}	λ_{rel}	λ_{hn}	RPR	μ AP	μ APsn
10	1	-	0.27	17.0	32.5
1	1	-	0.40	33.4	52.4
1	10	-	0.55	47.2	61.0
1	10	1	0.62	47.3	61.4
1	10	3	0.98	47.1	62.1
1	10	5	1.0	52.9	65.7
1	10	above 5	-	collapse	collapse

We next conduct a step-by-step experiment with DINO as the teacher model (Table 7). Our approach reduces the descriptor size by a factor of 12 but produces a performance comparable to that of the teacher model. As illustrated in Discussion 5.1, dimensional collapse occurs when HN loss is not employed. Applying HN loss thus successfully improves performance and prevents dimensional collapse.

Table 7. Ablation study of using DINO as teacher model.

Method	Network	Size	RPR	μ AP	μAP_{SN}
DINO (Teacher)	ViT-B/8	1536	0.99	32.6	54.4
RSD	EN-B0	1536	0.88	10.0	35.2
RSD — MoCo-v2	EN-B0	128	0.57	20.7	40.0
RSD — MoCo-v2 — HN (RDCD)	EN-B0	128	1.0	32.1	52.1

6. Conclusion

In this paper, we present RDCD, a novel method for training lightweight networks with compact descriptors for image copy detection in a self-supervised manner. We demonstrate in a series of experiments that RDCD, which combines RSD with HN loss effectively prevents dimensional collapse in lightweight architectures, and achieves a competitive performance across various benchmarks. We believe that this approach offers significant advantages in search speed and scalability for multimedia applications.

References

- [1] Soroush Abbasi Koohpayegani, Ajinkya Tejankar, and Hamed Pirsiavash. Compress: Self-supervised learning by compressing representations. *Advances in Neural Information Processing Systems*, 33:12980–12992, 2020. 2, 3
- [2] Artem Babenko, Anton Slesarev, Alexandr Chigorin, and Victor Lempitsky. Neural codes for image retrieval. In Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part I 13, pages 584–599. Springer, 2014. 1
- [3] Maxim Berman, Hervé Jégou, Andrea Vedaldi, Iasonas Kokkinos, and Matthijs Douze. Multigrain: a unified image embedding for classes and instances, 2019. 5
- [4] Cristian Buciluă, Rich Caruana, and Alexandru Niculescu-Mizil. Model compression. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 535–541, 2006. 3
- [5] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In Proceedings of the IEEE/CVF international conference on computer vision, pages 9650–9660, 2021. 1, 5
- [6] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020. 1, 2, 6
- [7] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020. 2, 4, 6
- [8] Matthijs Douze, Giorgos Tolias, Ed Pizzi, Zoë Papakipos, Lowik Chanussot, Filip Radenovic, Tomas Jenicek, Maxim Maximov, Laura Leal-Taixé, Ismail Elezi, et al. The 2021 image similarity dataset and challenge. arXiv preprint arXiv:2106.09672, 2021. 1, 2
- [9] Matthijs Douze, Giorgos Tolias, Ed Pizzi, Zoë Papakipos, Lowik Chanussot, Filip Radenovic, Tomas Jenicek, Maxim Maximov, Laura Leal-Taixé, Ismail Elezi, et al. The 2021 image similarity dataset and challenge, 2021. 1, 5
- [10] Zhiyuan Fang, Jianfeng Wang, Lijuan Wang, Lei Zhang, Yezhou Yang, and Zicheng Liu. Seed: Self-supervised distillation for visual representation. In *International Conference on Learning Representations*. 2, 3
- [11] Pierre Fernandez, Matthijs Douze, Hervé Jégou, and Teddy Furon. Active image indexing. *arXiv preprint arXiv:2210.10620*, 2022. 3
- [12] Yuting Gao, Jia-Xin Zhuang, Shaohui Lin, Hao Cheng, Xing Sun, Ke Li, and Chunhua Shen. Disco: Remedy selfsupervised learning on lightweight models with distilled contrastive learning, 2021. 2
- [13] Jindong Gu, Wei Liu, and Yonglong Tian. Simple distillation baselines for improving small self-supervised models. arXiv preprint arXiv:2106.11304, 2021.
- [14] Ruiqi Guo, Philip Sun, Erik Lindgren, Quan Geng, David Simcha, Felix Chern, and Sanjiv Kumar. Accelerating largescale inference with anisotropic vector quantization. In *International Conference on Machine Learning*, pages 3887–3896. PMLR, 2020. 1

- [15] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF con*ference on computer vision and pattern recognition, pages 9729–9738, 2020. 1
- [16] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. 2015. 3
- [17] Herve Jegou, Matthijs Douze, and Cordelia Schmid. Hamming embedding and weak geometry consistency for large scale image search-extended version. 2008. 5
- [18] Li Jing, Pascal Vincent, Yann LeCun, and Yuandong Tian. Understanding dimensional collapse in contrastive selfsupervised learning, 2021. 7
- [19] Jeff Johnson, Matthijs Douze, and Hervé Jégou. Billionscale similarity search with gpus. *IEEE Transactions on Big Data*, 7(3):535–547, 2019.
- [20] Herve Jégou, Matthijs Douze, and Cordelia Schmid. Product quantization for nearest neighbor search. *IEEE Transactions* on Pattern Analysis and Machine Intelligence, 33(1):117– 128, 2011. 3
- [21] Giorgos Kordopatis-Zilos, Giorgos Tolias, Christos Tzelepis, Ioannis Kompatsiaris, Ioannis Patras, and Symeon Papadopoulos. Self-supervised video similarity learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 4756–4766, 2023. 5
- [22] Ting Liu, Charles Rosenberg, and Henry A Rowley. Clustering billions of images with large scale nearest neighbor search. In 2007 IEEE workshop on applications of computer vision (WACV'07), pages 28–28. IEEE, 2007.
- [23] Hao Luo, Youzhi Gu, Xingyu Liao, Shenqi Lai, and Wei Jiang. Bag of tricks and a strong baseline for deep person re-identification. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops, 2019. 2
- [24] Eva Mohedano, Kevin McGuinness, Noel E O'Connor, Amaia Salvador, Ferran Marques, and Xavier Giró-i Nieto. Bags of local convolutional features for scalable instance search. In *Proceedings of the 2016 ACM on international* conference on multimedia retrieval, pages 327–331, 2016. 1
- [25] Hyeonwoo Noh, Andre Araujo, Jack Sim, Tobias Weyand, and Bohyung Han. Large-scale image retrieval with attentive deep local features. In *Proceedings of the IEEE international* conference on computer vision, pages 3456–3465, 2017.
- [26] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding, 2018.
- [27] Sergio Manuel Papadakis and Sanjay Addicam. Producing augmentation-invariant embeddings from real-life imagery. arXiv preprint arXiv:2210.10620, 2021. 2
- [28] Wonpyo Park, Dongju Kim, Yan Lu, and Minsu Cho. Relational knowledge distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3967–3976, 2019.
- [29] Nikolaos Passalis and Anastasios Tefas. Learning deep representations with probabilistic knowledge transfer. In Proceedings of the European Conference on Computer Vision (ECCV), pages 268–284, 2018. 3

- [30] Florent Perronnin, Yan Liu, and Jean-Michel Renders. A family of contextual measures of similarity between distributions with application to image retrieval. In 2009 IEEE Conference on computer vision and pattern recognition, pages 2358–2365. IEEE, 2009. 5
- [31] Ed Pizzi, Sreya Dutta Roy, Sugosh Nagavara Ravindra, Priya Goyal, and Matthijs Douze. A self-supervised descriptor for image copy detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14532–14542, 2022. 1, 2, 5, 6
- [32] Filip Radenović, Giorgos Tolias, and Ondřej Chum. Finetuning cnn image retrieval with no human annotation. *IEEE transactions on pattern analysis and machine intelligence*, 41(7):1655–1668, 2018. 1
- [33] Ali S Razavian, Josephine Sullivan, Stefan Carlsson, and Atsuto Maki. Visual instance retrieval with deep convolutional networks. ITE Transactions on Media Technology and Applications, 4(3):251–258, 2016. 1
- [34] Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. Fitnets: Hints for thin deep nets, 2014. 3
- [35] Alexandre Sablayrolles, Matthijs Douze, Cordelia Schmid, and Hervé Jégou. Spreading vectors for similarity search, 2018. 2
- [36] Haizhou Shi, Youcai Zhang, Siliang Tang, Wenjie Zhu, Yaqian Li, Yandong Guo, and Yueting Zhuang. On the efficacy of small self-supervised contrastive models without distillation signals. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 2225–2234, 2022.
- [37] Oriane Siméoni, Yannis Avrithis, and Ondrej Chum. Local features and visual words emerge in activations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11651–11660, 2019.
- [38] Sivic and Zisserman. Video google: a text retrieval approach to object matching in videos. In *Proceedings Ninth IEEE International Conference on Computer Vision*, pages 1470–1477 vol.2, 2003. 3
- [39] Ajinkya Tejankar, Soroush Abbasi Koohpayegani, Vipin Pillai, Paolo Favaro, and Hamed Pirsiavash. Isd: Self-supervised learning by iterative similarity distillation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9609–9618, 2021. 3
- [40] Bart Thomee, David A Shamma, Gerald Friedland, Benjamin Elizalde, Karl Ni, Douglas Poland, Damian Borth, and Li-Jia Li. Yfcc100m: The new data in multimedia research. *Communications of the ACM*, 59(2):64–73, 2016. 5
- [41] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive representation distillation, 2019. 3
- [42] Giorgos Tolias, Tomas Jenicek, and Ondřej Chum. Learning and aggregating deep local descriptors for instance-level recognition. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16*, pages 460–477. Springer, 2020. 1
- [43] Frederick Tung and Greg Mori. Similarity-preserving knowledge distillation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 1365–1374, 2019.

- [44] Wenhao Wang, Yifan Sun, and Yi Yang. A benchmark and asymmetrical-similarity learning for practical image copy detection, 2023. 5
- [45] Wenhao Wang, Weipu Zhang, Yifan Sun, and Yi Yang. Bag of tricks and a strong baseline for image copy detection. *arXiv preprint arXiv:2111.08004*, 2021. 1
- [46] Junho Yim, Donggyu Joo, Jihoon Bae, and Junmo Kim. A gift from knowledge distillation: Fast optimization, network minimization and transfer learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4133–4141, 2017. 3
- [47] Shuhei Yokoo. Contrastive learning with large memory bank and negative embedding subtraction for accurate copy detection. arXiv preprint arXiv:2112.04323, 2021. 2
- [48] Sergey Zagoruyko and Nikos Komodakis. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer, 2016. 3
- [49] Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. Barlow twins: Self-supervised learning via redundancy reduction. In *International conference on machine learning*, pages 12310–12320. PMLR, 2021. 1