

Online Backfilling with No Regret for Large-Scale Image Retrieval

Seonguk Seo^{1,†} Mustafa Gokhan Uzunbas³ Bohyung Han^{1,2}
Sara Cao³ Joena Zhang³ Taipeng Tian³ Ser-Nam Lim³

¹ECE & ^{1,2}IPAI, Seoul National University ³Meta AI

{seonguk, bhhan}@snu.ac.kr

{gokhanuzunbas, joenazhang, xuefeicao01, ttp, sernamlim}@meta.com

Abstract

Backfilling is the process of re-extracting all gallery embeddings from upgraded models in image retrieval systems. It inevitably requires a prohibitively large amount of computational cost and even entails the downtime of the service. Although backward-compatible learning sidesteps this challenge by tackling query-side representations, this leads to suboptimal solutions in principle because gallery embeddings cannot benefit from model upgrades. We address this dilemma by introducing an online backfilling algorithm, which enables us to achieve a progressive performance improvement during the backfilling process while not sacrificing the final performance of new model after the completion of backfilling. To this end, we first propose a simple distance rank merge technique for online backfilling. Then, we incorporate a reverse transformation module for more effective and efficient merging, which is further enhanced by adopting a metric-compatible contrastive learning approach. These two components help to make the distances of old and new models compatible, resulting in desirable merge results during backfilling with no extra computational overhead. Extensive experiments show the effectiveness of our framework on four standard benchmarks in various settings.

1. Introduction

Image retrieval models [5, 10, 21, 23] have achieved remarkable performance by adopting deep neural networks for representing images. Yet, all models need to be upgraded at times to take advantage of improvements in training datasets, network architectures, and training techniques. This unavoidably leads to the need for re-extracting the features from millions or even billions of gallery images using the upgraded new model. This process, called *backfilling*

or *re-indexing*, needs to be completed before the retrieval system can benefit from the new model, which may take months in practice.

To sidestep this bottleneck, several backfilling-free approaches based on backward-compatible learning [4, 13, 19, 20, 22] have been proposed. They learn a new model while ensuring that its feature space is still compatible with the old one, thus avoiding the need for updating old gallery embeddings. Although these approaches have achieved substantial performance gains without backfilling, they achieve feature compatibility at the expense of feature discriminability and their performance is suboptimal. We argue that backward-compatible learning is not a fundamental solution and backfilling is still essential to accomplish state-of-the-art performance without performance sacrifices.

To resolve this compatibility-discriminability dilemma, we relax the backfill-free constraint and propose a novel online backfilling algorithm equipped with three technical components. We posit that an online backfilling technique needs to satisfy three essential conditions: 1) immediate deployment after the completion of model upgrade, 2) progressive and non-trivial performance gains in the middle of backfilling, and 3) no degradation of final performance compared to offline backfilling. To this end, we first propose a distance rank merge framework to make online backfilling feasible, which retrieves images from both the old and new galleries separately and merge their results to obtain the final retrieval outputs even when backfilling is still ongoing. While this approach provides a monotonic performance increase with the progress of backfilling regardless of the gallery of interest and network architectures, it requires feature computations twice, once from the old model and another from the new one at the inference stage of a query. To overcome this limitation, we introduce a reverse transformation module, which is a lightweight mapping network between the old and new embeddings. The reverse transformation module allows us to obtain the query representations compatible with both the old and new galleries

[†] This work was mostly done during an internship at Meta AI.

using only a single feature extraction. On the other hand, however, we notice that the scales of distance in the embedding spaces of the two models could be significantly different. We resolve the limitation with a metric compatible learning technique, which calibrates the distances of two models via contrastive learning, further enhancing performance of rank merge.

The main contributions of our work are summarized as follows.

- We propose an online backfilling approach, a fundamental solution for model upgrades in image retrieval systems, based on distance rank merge to overcome the compatibility-discriminability dilemma in existing compatible learning methods.
- We incorporate a reverse query transform module to make it compatible with both the old and new galleries while computing the feature extraction of query only once in the middle of the backfilling process.
- We adopt a metric-compatible learning technique to make the merge process robust by calibrating distances in the feature embedding spaces given by the old and new models.
- The proposed approach outperforms all existing methods by significant margins on four standard benchmark datasets under various scenarios.

The rest of this paper is organized as follows. Section 2 reviews the related works. We present the main framework of online backfilling in Section 3, and discuss the technical components for improvement in Section 4 and 5. We demonstrate the effectiveness of the proposed framework in Section 6 and conclude this paper in Section 7.

2. Related Work

Backward compatible learning Backward compatibility refers to the property to support older versions in hardware or software systems. It has been recently used in model upgrade scenarios in image retrieval systems. Since the feature spaces given by the models relying on training datasets in different regimes are not compatible [11, 24], model upgrades require re-extraction of all gallery images from new models, which takes a huge amount of computational cost. To prevent this time-consuming backfilling cost, backward compatible training (BCT) [1, 13, 15, 19, 22, 26] has been proposed to learn better feature representations while being compatible with old embeddings, which makes the new model backfill-free. Shen *et al.* [19] employ the influence loss that utilizes the old classifier as a regularizer when training the new model. LCE [13] introduces an alignment

loss to align the class centers between old and new models and a boundary loss that restricts more compact intra-class distributions for the new model. Bai *et al.* [1] propose a joint prototype transfer with structural regularization to align two embedding features. UniBCT [26] presents a structural prototype refinement algorithm that first refines noisy old features with graph transition and then conducts backward compatible training. Although these approaches improved compatible performance without backfilling, they clearly sacrifice feature discriminability to achieve feature compatibility with non-ideal old gallery embeddings.

Compatible learning with backfilling To overcome the inherent limitation of backward compatible learning, several approaches [17, 20, 25] have been proposed to utilize backfilling but efficiently. Forward compatible training (FCT) [17] learn a lightweight transformation module that updates old gallery embeddings to be compatible with new embeddings. Although it gives better compatible performance than BCT, it requires an additional side-information [2] to map from old to new embeddings, which limits its practicality. Moreover, FCT still suffers from computational bottleneck until all old gallery embeddings are transformed, especially when the side-information needs to be extracted. On the other hand, RACT [25] and BiCT [20] alleviate this bottleneck issue by backfilling the gallery embeddings in an online manner. RACT first trains a backward-compatible new model with regression-alleviating loss, then backfills the old gallery embeddings with the new model. Because the new feature space is compatible with the old one, the new model can be deployed right away while backfilling is carried out in the background. BiCT further reduces the backfilling cost by transforming the old gallery embeddings with forward-compatible training [17]. Although both approaches can utilize online backfilling, they still sacrifice the final performance because the final new embeddings are constrained by the old ones. Unlike these methods, our framework enables online backfilling while fully exploiting the final new model performance without any degradation.

3. Image Retrieval by Rank Merge

This section discusses our baseline image retrieval algorithm that makes online backfilling feasible. We first present our motivation and then describe technical details with empirical observations.

3.1. Overview

Our goal is to develop a fundamental solution via online backfilling to overcome the compatibility-discriminability trade-off in compatible model upgrade. This strategy removes inherent limitations of backfill-free backward-compatible learning—the inability to use state-of-the-

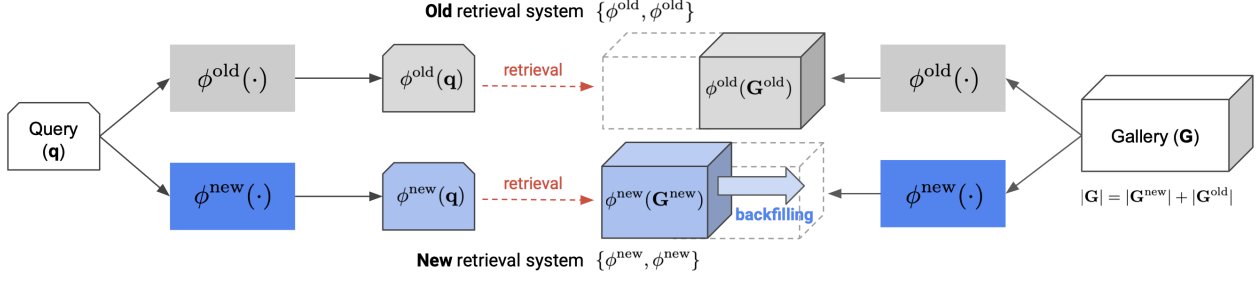


Figure 1. Image retrieval with the proposed distance rank merge technique. In the middle of backfilling, we retrieve images independently using two separate models and their galleries, and merge the retrieval results based on their distances. Note that the total number of gallery embeddings are fixed throughout the backfilling process, *i.e.*, $|G| = |G^{new}| + |G^{old}|$.

art representations of gallery images through model upgrades—while avoiding prohibitive costs, including the situation that we cannot benefit from model upgrade of the offline backfilling process, until backfilling is completed. To be specific, the proposed image retrieval system with online backfilling should satisfy the following three conditions:

1. The system can be deployed immediately as soon as model upgrade is complete.
2. The performance should monotonically increase without negative flips¹ as backfill progresses.
3. The final performance should not be sacrificed compared to the algorithm relying on offline backfilling.

We present a distance rank merge approach for image retrieval, which enables online backfilling in arbitrary model upgrade scenarios. Our method maintains two separate retrieval pipelines corresponding to the old and new models and merges the retrieval results from the two models based on distances from a query embedding. This allows us to run the retrieval system without a warm-up period and achieve surprisingly good results during the backfill process. Note that the old and new models are not required to be compatible at this moment but we will make them so to further improve performance in the subsequent sections.

3.2. Formulation

Let $q \in Q$ be a query image and $G = \{g_1, \dots, g_N\}$ be a gallery composed of N images. An embedding network $\phi(\cdot)$ projects an image onto a learned feature embedding space. To retrieve the closest gallery image given a query, we find $\arg \min_{g \in G} \text{dist}(\phi(q), \phi(g))$, where $\text{dist}(\cdot, \cdot)$ is a distance metric. Following [19], we define the retrieval performance as

$$\mathcal{M}(\phi(Q), \phi(G)), \quad (1)$$

¹The “negative flip” refers to performance degradation caused by incorrect retrievals of samples by the new model, which were correctly recognized by the old model.

where $\mathcal{M}(\cdot, \cdot)$ is an evaluation metric such as mean average precision (mAP) or cumulative matching characteristics (CMC), and $\phi(\cdot)$ indicates embedding models for query and gallery, respectively.

Backward compatibility Denote the old and new embedding networks by $\phi^{old}(\cdot)$ and $\phi^{new}(\cdot)$ respectively. If $\phi^{new}(\cdot)$ is backward compatible with $\phi^{old}(\cdot)$, then we can perform search on a set of old gallery embeddings using a new query embedding, *i.e.*, $\arg \min_{g \in G} \text{dist}(\phi^{new}(q), \phi^{old}(g))$. As stated in [19], the backward compatibility is achieved when the following criterion is satisfied:

$$\mathcal{M}(\phi^{new}(Q), \phi^{old}(G)) > \mathcal{M}(\phi^{old}(Q), \phi^{old}(G)). \quad (2)$$

From now, we refer to a pair of embedding networks for query and gallery as a retrieval system, *e.g.*, $\{\phi(\cdot), \phi(\cdot)\}$.

Rank merge Assume that the first M out of a total of N images are backfilled, *i.e.*, $G^{new} = \{g_1, \dots, g_M\}$ and $G^{old} = \{g_{M+1}, \dots, g_N\}$. Note that the total number of stored gallery embeddings is fixed to N during the backfilling process, *i.e.*, $G^{old} = G - G^{new}$. Then, we first conduct image retrieval using the individual retrieval systems, $\{\phi^{old}, \phi^{old}\}$ and $\{\phi^{new}, \phi^{new}\}$, independently as

$$g_m = \arg \min_{g_i \in G^{old}} \text{dist}(\phi^{old}(q), \phi^{old}(g_i)), \quad (3)$$

$$g_n = \arg \min_{g_j \in G^{new}} \text{dist}(\phi^{new}(q), \phi^{new}(g_j)). \quad (4)$$

Figure 1 illustrates the retrieval process. For each query image q , we finally select g_m if $\text{dist}(\phi^{old}(q), \phi^{old}(g_m)) < \text{dist}(\phi^{new}(q), \phi^{new}(g_n))$ and g_n otherwise. The retrieval performance after rank merge during backfilling is given by

$$\mathcal{M}_t := \mathcal{M}(\{\phi^{old}(Q), \phi^{new}(Q)\}, \{\phi^{old}(G_t^{old}), \phi^{new}(G_t^{new})\}), \quad (5)$$

where $t \in [0, 1]$ indicates the rate of backfilling completion, *i.e.*, $|G_t^{new}| = t|G|$ and $|G_t^{old}| = (1 - t)|G|$. The criteria

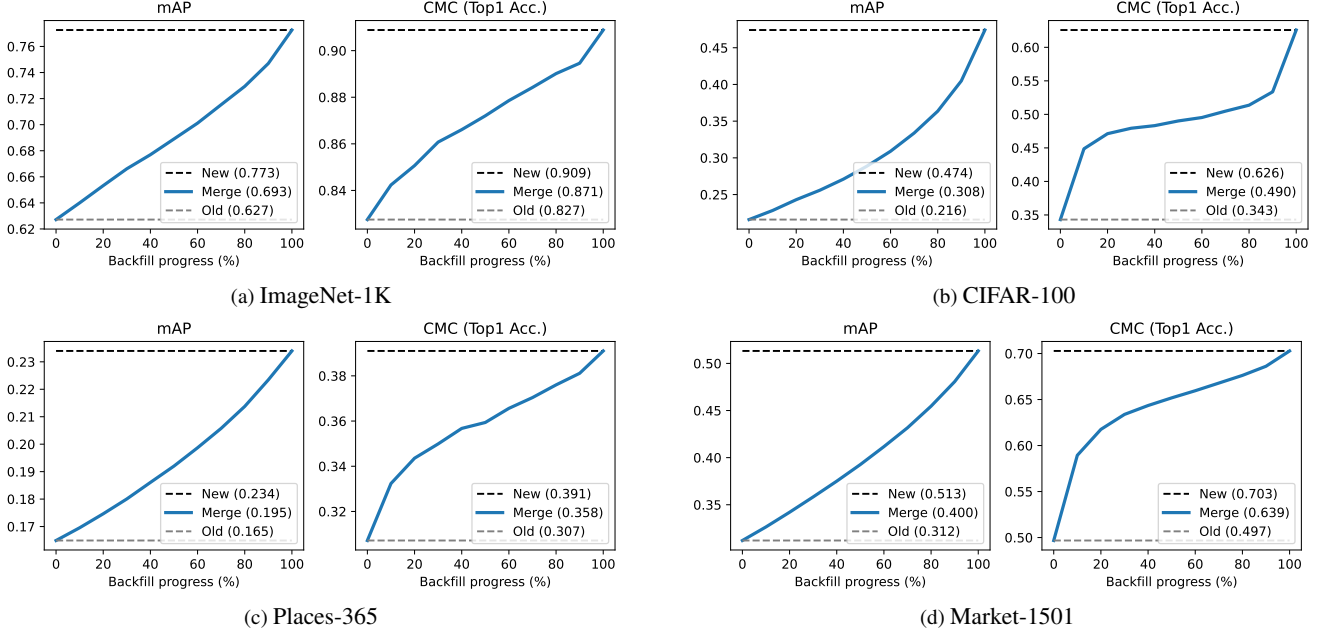


Figure 2. mAP and CMC results on the standard benchmarks using ResNet-18. *Old* and *New* denote the performance without backfilling and with offline backfilling, respectively. The proposed distance rank merging of the old and new models, denoted by *Merge*, exhibits desirable results; the performance monotonically increases as backfill progresses without negative flips for all datasets and our algorithm based on online backfilling achieves competitive final performances with offline backfilling. The numbers in the legend indicate either AUC_{mAP} or AUC_{CMC} scores.

discussed in Section 3.1 are formally defined as

$$\mathcal{M}_0 \geq \mathcal{M}(\phi^{\text{old}}(\mathbf{Q}), \phi^{\text{old}}(\mathbf{G})), \quad (6)$$

$$\mathcal{M}_1 \geq \mathcal{M}(\phi^{\text{new}}(\mathbf{Q}), \phi^{\text{new}}(\mathbf{G})), \quad (7)$$

$$\mathcal{M}_{t_1} \geq \mathcal{M}_{t_2} \text{ if } t_1 \geq t_2. \quad (8)$$

Comprehensive evaluation To measure both backfilling cost and model performance comprehensively during online backfilling, we utilize the following metrics that calculate the area under mAP or CMC curves as

$$AUC_{mAP} = \int_0^1 mAP_t dt \text{ and } AUC_{CMC} = \int_0^1 CMC_t dt.$$

3.3. Merge Results

We present the results from the rank merge strategy on two standard benchmarks, including ImageNet-1K [18] and Places-365 [28], in Figure 2. Our rank merging approach yields strong and robust results for all datasets; both mAP and CMC monotonically increase without negative flips as backfill progresses even though the old and new models are not compatible each other. Also, it takes full advantage of the new model until the end of backfilling without suffering from performance degradation. This validates that our rank merge technique satisfies the criteria for online backfilling discussed in Section 3.1 and 3.2. Please refer to Section 6.1 for the experimental detail.

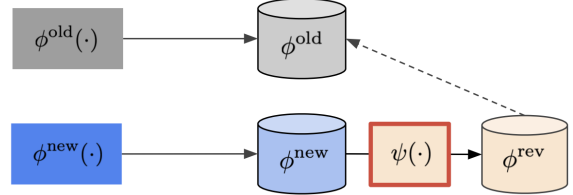


Figure 3. Reverse query transform module, $\psi(\cdot)$, learns a mapping from new to old feature spaces. We only update the parameters of the module $\psi(\cdot)$ (in red rectangle) during training.

4. Reverse Query Transform

Our baseline image retrieval method is model-agnostic, free from extra training, and effective for performance improvement. However, one may argue that the proposed approach is computationally expensive at inference time because we need to conduct feature extraction twice per query for both the old and new models. This section discusses how to alleviate this limitation by introducing a small network, called the reverse query transform module.

4.1. Basic Formulation

To reduce the computational cost incurred by computing query embeddings twice at the inference stage, we compute the embedding using the new model and transform it to the version compatible with the old model through the reverse

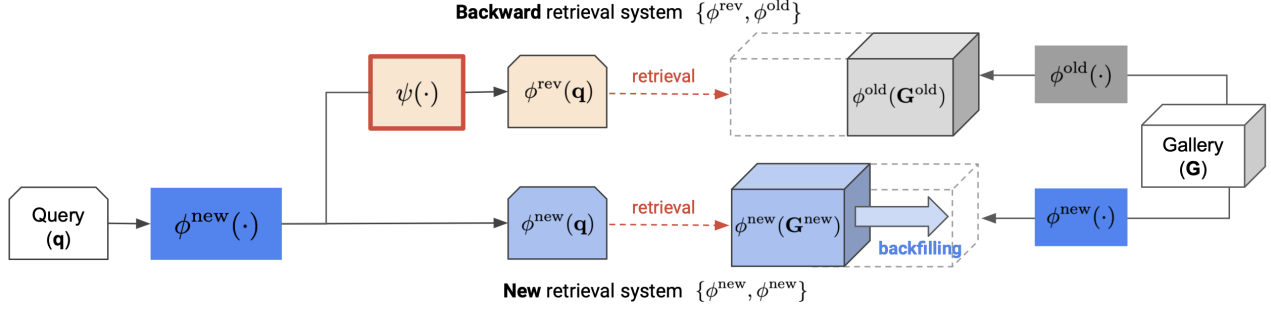


Figure 4. Image retrieval merging with reverse query transform module. Backward retrieval system consists of reversely transformed new query and old gallery, $\{\phi^{rev}, \phi^{old}\}$. The final image retrieval results are given by merging the outputs from $\{\phi^{rev}, \phi^{old}\}$ and $\{\phi^{new}, \phi^{new}\}$.

query transform module as illustrated in Figure 3. To establish such a mechanism, we fix the parameters of the old and new models $\{\phi^{old}, \phi^{new}\}$ after training them independently, and train a lightweight network, $\psi(\cdot)$, which transforms the embedding in the new model to the one in the old model. For each training example $\mathbf{x}$, our objective is minimizing the following loss:

$$\mathcal{L}_{\text{RQT}}(\mathbf{x}) := \text{dist}(\psi(\phi^{new}(\mathbf{x})), \phi^{old}(\mathbf{x})), \quad (9)$$

where $\text{dist}(\cdot, \cdot)$ is a distance metric such as ℓ_2 or cosine distances. Because we only update the parameters in $\psi(\cdot)$, not the ones in $\phi^{new}(\cdot)$ or $\phi^{old}(\cdot)$, we can still access the representations given by the new model at no cost even after the optimization of $\psi(\cdot)$. Note that this reverse query transform module differs from FCT [17] mainly in terms of transformation direction and requirement of side information. FCT performs a transformation from the old representation to the new, while the opposite is true for our proposed approach. Since the embedding quality of a new model is highly likely to be better than that of an old one, our reverse transformation module performs well even without additional side information and, consequently, is more practical and efficient.

4.2. Integration into Baseline Retrieval System

Figure 4 illustrates the distance rank merge process together with the proposed reverse transformation module. The whole procedure consists of two retrieval systems defined by a pair of query and gallery representations, backward retrieval system $\{\phi^{rev}, \phi^{old}\}$ and new retrieval system $\{\phi^{new}, \phi^{new}\}$, where $\phi^{rev} := \psi(\phi^{new})$. Note that we obtain both the new and compatible query embeddings, $\phi^{new}(\mathbf{q})$ and $\phi^{rev}(\mathbf{q}) = \psi(\phi^{new}(\mathbf{q}))$, using a shared feature extraction network, $\phi^{new}(\cdot)$.

The entire image retrieval pipeline consists of two parts: 1) feature extraction of a query image and 2) search for the nearest image in a gallery from the query. Compared to the image retrieval based on a single model, the computational cost of the proposed model with rank merge requires negligible additional cost, which corresponds to feature transformation $\psi(\cdot)$ in the first part. Note that the number of total

gallery embeddings is fixed, *i.e.*, $|\mathbf{G}^{new}| + |\mathbf{G}^{old}| = |\mathbf{G}|$, so the cost of the second part is always the same in both cases.

5. Distance Calibration

While the proposed rank merge technique with the basic reverse transformation module works well, there exists room for improvement in calibrating feature embedding spaces of both systems. This section discusses the issues in details and presents how we figure them out.

5.1. Cross-Model Contrastive Learning

The objective in (9) cares about the positive pairs ϕ^{old} and ϕ^{rev} with no consideration of negative pairs, which can sometimes lead to misranked position. To handle this issue, we employ a supervised contrastive learning loss [7, 14] to consider both positive and negative pairs as follows:

$$\mathcal{L}_{\text{CL}}(\mathbf{x}_i, y_i) = -\log \frac{\sum_{y_k=y_i} s_{ik}^{old}}{\sum_{y_k=y_i} s_{ik}^{old} + \sum_{y_k \neq y_i} s_{ik}^{old}}, \quad (10)$$

where $s_{ij}^{old} = \exp(-\text{dist}(\phi^{rev}(\mathbf{x}_i), \phi^{old}(\mathbf{x}_j)))$ and y_i denotes the class membership of the i^{th} sample. For more robust contrastive training, we perform hard example mining for both the positive and negative pairs². Such a contrastive learning approach facilitates distance calibration and improves feature discrimination because it promotes separation of the positive and negative examples.

Now, although the distances within the backward retrieval system $\{\phi^{rev}, \phi^{old}\}$ become more comparable, they are still not properly calibrated in terms of the distances in the new retrieval system $\{\phi^{new}, \phi^{new}\}$. Considering distances in both retrieval systems jointly when we train the reverse transformation module, we can obtain more comparable distances and consequently achieve more reliable rank merge results. From this perspective, we propose a

²For each anchor, we select the half of the examples in each of positive and negative labels based on the distances from the anchor.

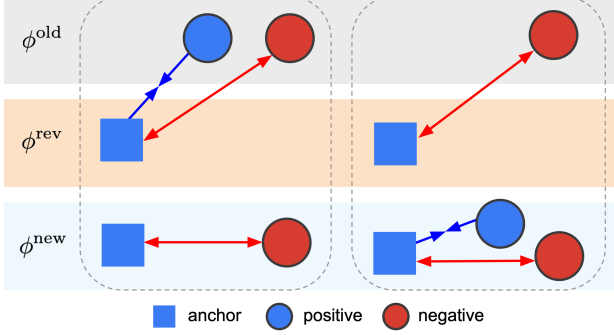


Figure 5. Illustration of cross-model contrastive learning loss with backward retrieval system $\{\phi^{old}, \phi^{rev}\}$ and new retrieval system $\{\phi^{new}, \phi^{new}\}$. Two boxes with dotted lines corresponds to two terms in (11). For each retrieval system, the distances between positive pairs are learned to be both smaller than those of negative pairs in the two retrieval systems.

cross-model contrastive learning loss as

$$\mathcal{L}_{CMCL}(\mathbf{x}_i, y_i) = \begin{aligned} & -\log \frac{\sum_{y_k=y_i} s_{ik}^{old}}{\sum_{y_k=y_i} s_{ik}^{old} + \sum_{y_k \neq y_i} s_{ik}^{old} + \sum_{y_k \neq y_i} s_{ik}^{new}} \\ & -\log \frac{\sum_{y_k=y_i} s_{ik}^{new}}{\sum_{y_k=y_i} s_{ik}^{new} + \sum_{y_k \neq y_i} s_{ik}^{new} + \sum_{y_k \neq y_i} s_{ik}^{old}}, \end{aligned} \quad (11)$$

where $s_{ij}^{new} = \exp(-\text{dist}(\phi^{new}(\mathbf{x}_i), \phi^{new}(\mathbf{x}_j)))$ and $s_{ij}^{old} = \exp(-\text{dist}(\phi^{rev}(\mathbf{x}_i), \phi^{old}(\mathbf{x}_j)))$. Figure 5 illustrates the concept of the loss function. The positive pairs from the backward retrieval system $\{\phi^{rev}, \phi^{old}\}$ are trained to locate closer to the anchor than not only the negative pairs from the same system but also the ones from the new system $\{\phi^{new}, \phi^{new}\}$, and vice versa. We finally replace (9) with (11) for training the reverse transformation module. Compared to (10), additional heterogeneous negative terms in the denominator of (11) play a role as a regularizer to make the distances from one model directly comparable to those from other one, which is desirable for our rank merge strategy.

5.2. Training New Feature Embedding

Until now, we do not jointly train the reverse transformation module $\psi(\cdot)$ and the new feature extraction module $\phi^{new}(\cdot)$ as illustrated in Figure 3. This hampers the compatibility between the backward and new retrieval systems because the backward retrieval system $\{\phi^{rev}, \phi^{old}\}$ is the only part to be optimized while the new system $\{\phi^{new}, \phi^{new}\}$ is fixed. To provide more flexibility, we add another transformation module $\rho(\cdot)$ on top of the new model as shown in Figure 6, where $\rho^{new} = \rho(\phi^{new})$ and $\rho^{rev} = \psi(\rho(\phi^{new}))$. In this setting, we use ρ^{new} as the final new model instead of ϕ^{new} , and our rank merge process employs $\{\rho^{rev}, \phi^{old}\}$ and

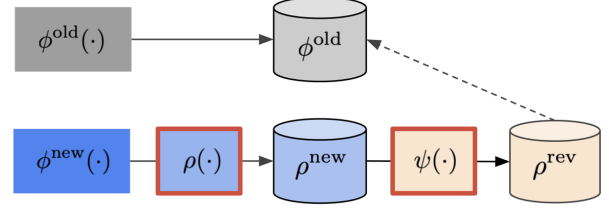


Figure 6. Compatible training with learnable new embedding. Compared to Figure 3, another transformation module $\rho(\cdot)$ is incorporated on top of the new model to learn new embedding favorable to our rank merging. The retrieval results are now merged from $\{\rho^{rev}, \phi^{old}\}$ and $\{\rho^{new}, \rho^{new}\}$.

$\{\rho^{new}, \rho^{new}\}$ eventually. This strategy helps to achieve a better compatibility by allowing both systems to be trainable.

The final function to train the reverse transformation module has the identical form to $\mathcal{L}_{CMCL}$ in (11) except for the definitions of s_{ij}^{new} and s_{ij}^{old} , which are given by

$$s_{ij}^{new} = \exp(-\text{dist}(\rho^{new}(\mathbf{x}_i), \rho^{new}(\mathbf{x}_j))) \quad (12)$$

$$s_{ij}^{old} = \exp(-\text{dist}(\rho^{rev}(\mathbf{x}_i), \phi^{old}(\mathbf{x}_j))). \quad (13)$$

Note that this extension does not result in computational overhead at inference stage yet improves the performance even further.

6. Experiments

We present our experiment setting, the performance of the proposed approach, and results from the analysis of algorithm characteristics.

6.1. Dataset and Evaluation Protocol

We employ four standard benchmarks, which includes ImageNet-1K [18], CIFAR-100 [9], Places-365 [28], Market-1501 [27]. As in previous works [17, 19], we adopt the extended-class setting in model upgrade; the old model is trained with examples from a half of all classes while the new model is trained with all samples. For example, on the ImageNet-1K dataset, the old model is trained with the first 500 classes and the new model is trained with the whole 1,000 classes.

Following the previous works [17, 20, 25], we measure mean average precision (mAP) and cumulative matching characteristics (CMC)³. We also report our comprehensive results in terms of AUC_{mAP} and AUC_{CMC} at 10 backfill time slices, *i.e.*, $t \in \{0.0, 0.1, \dots, 1.0\}$ in (5).

6.2. Implementation Details

We employ ResNet-18 [6], ResNet-50 [6], and ViT-B/32 [3] as our backbone architectures for either old or new

³CMC corresponds to top- k accuracy, and we report top-1 accuracy in all tables and graphs.

Table 1. Comparison with existing compatible learning methods on four standard benchmarks in homogeneous model upgrades. *Gain* denotes relative gain that each method achieves from old model in terms of AUC_{mAP} , compared to the gain of new model. The proposed framework, dubbed as RM, consistently outperforms all other models with significantly large margins for all datasets. Note that RM_{naive} indicates the basic version of distance rank merge described in Sec. 3.2 and that *Old* and *New* denote embedding models of gallery images.

	ImageNet-1K			CIFAR-100			Places-365			Market-1501		
	AUC_{mAP}	AUC_{CMC}	Gain	AUC_{mAP}	AUC_{CMC}	Gain	AUC_{mAP}	AUC_{CMC}	Gain	AUC_{mAP}	AUC_{CMC}	Gain
Old	31.2	49.7	0%	21.6	34.3	0%	16.5	30.7	0%	62.7	82.7	0%
New	51.3	70.3	100%	47.4	62.6	100%	23.4	39.1	100%	77.3	90.9	100%
RM_{naive} (Ours)	40.0	63.9	44%	30.8	49.1	36%	19.5	35.8	43%	69.2	87.0	45%
BCT [19]	32.0	46.3	4%	26.4	43.5	19%	17.5	37.0	14%	66.6	84.3	27%
FCT [17]	36.9	58.7	28%	27.1	49.4	21%	22.5	37.3	87%	66.4	84.2	25%
FCT (w/ side-info) [17]	43.6	65.0	62%	37.0	53.9	60%	23.7	38.3	104%	66.4	84.4	25%
BiCT [20]	35.1	59.7	19%	29.0	48.3	29%	19.0	34.9	36%	65.0	82.4	16%
RM (Ours)	53.4	68.1	110%	41.4	60.7	78%	28.2	41.7	170%	70.7	87.6	55%

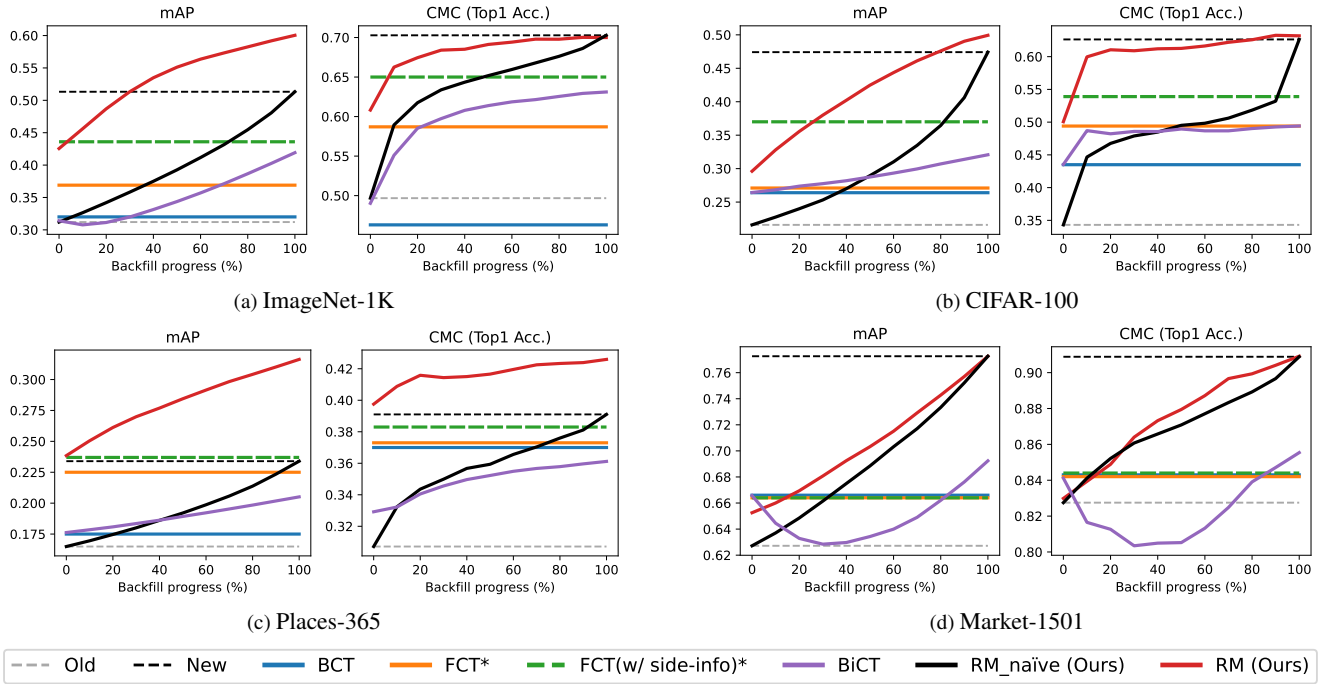


Figure 7. mAP and CMC (Top-1 Acc.) results of our full framework in comparison to existing approaches. The numbers in the legend indicate either AUC_{mAP} or AUC_{CMC} scores.

models. All transformation modules, $\psi(\cdot)$ and $\rho(\cdot)$, consist of 1 to 5 linear layer blocks, where each block is composed of a sequence of operations, (Linear $\rightarrow$ BatchNorm $\rightarrow$ ReLU), except for the last block that only has a Linear layer. Our algorithm does not use any side-information. Our modules are trained with the Adam optimizer [8] for 50 epoch, where the learning rate is 1×10^{-4} at the beginning and decayed using cosine annealing [12]. Our frameworks are implemented with the Pytorch [16] library and we plan to release the source codes of our work.

6.3. Results

Homogeneous model upgrade We present the quantitative results in the homogeneous model upgrade scenario, where old and new models have the same architecture. We employ ResNet-50 for ImageNet and ResNet-18 for other datasets. Table 1 and Figure 7 compare the proposed framework, referred to as RM (Rank Merge), with existing compatible learning approaches, including BCT [19], FCT [17], and BiCT [20]. As shown in the table, RM consistently outperforms all the existing compatible learning methods by remarkably significant margins in all datasets. BCT [19] learns backward compatible feature representations, which is backfill-free, but its performance gain is not impressive.

FCT [17] achieves meaningful performance improvement by transforming old gallery features, but most of the gains come from side-information [2]. For example, if side-information is not available, the performance gain of FCT drops from 62% to 28% on the ImageNet dataset. Also, such side-information is not useful for the re-identification dataset, Market-1501, mainly because the model for the side-information is trained for image classification using the ImageNet dataset, which shows its limited generalizability. On the other hand, although BiCT [20] takes advantage of online backfilling with less backfilling cost, it suffers from degraded final performance and negative flips in the middle of backfilling. Note that RM_{naive} , our naïve rank merging between old and new models, is already competitive to other approaches.

Heterogeneous model upgrade We evaluate our framework in more challenging scenarios and present the results in Figure 8, where the old and new models have different architectures, *e.g.*, ResNet-18 $\rightarrow$ ResNet-50 or ResNet-18 $\rightarrow$ ViT-B/32. In this figure, RM_{RQT} (green line) denotes our ablative model trained with (9). Even in this setting, where both embedding spaces are more incompatible, our rank merge results from the old and new models still manage to achieve a monotonous performance growth curve and RM improves the overall performance significantly further, which validates the robustness of our frameworks.

Ablation study We analyze the results from the ablations of models for our cross-model contrastive learning. For compatible training, CL-S employs contrastive learning within the backward system only as in (10) while our CMCL considers distance metrics from both backward and new retrieval systems simultaneously as in (11). For a more thorough ablation study, we also design and test another metric learning objective, called CL-M, which is given by

$$\begin{aligned} \mathcal{L}_{CL-M}(x_i, y_i) = & -\log \frac{\sum_{y_k=y_i} s_{ik}^{old}}{\sum_{y_k=y_i} s_{ik}^{old} + \sum_{y_k \neq y_i} s_{ik}^{old}} \\ & -\log \frac{\sum_{y_k=y_i} s_{ik}^{new}}{\sum_{y_k=y_i} s_{ik}^{new} + \sum_{y_k \neq y_i} s_{ik}^{new}}, \quad (14) \end{aligned}$$

which conducts contrastive learning for both backward and new retrieval systems separately. Figure 9 visualizes the results from the ablation studies, where CMCL consistently outperforms both CL-S and CL-M in various datasets and architectures. CL-M generally gives better merge results than CL-S because it calibrates the distances of new retrieval system additionally. However, CL-M still suffers from negative flips because the distance metrics of both retrieval systems are calibrated independently and not learned to be directly comparable to each other. On the other hand, CMCL improves overall performance curves consistently without negative flips. This validates that con-

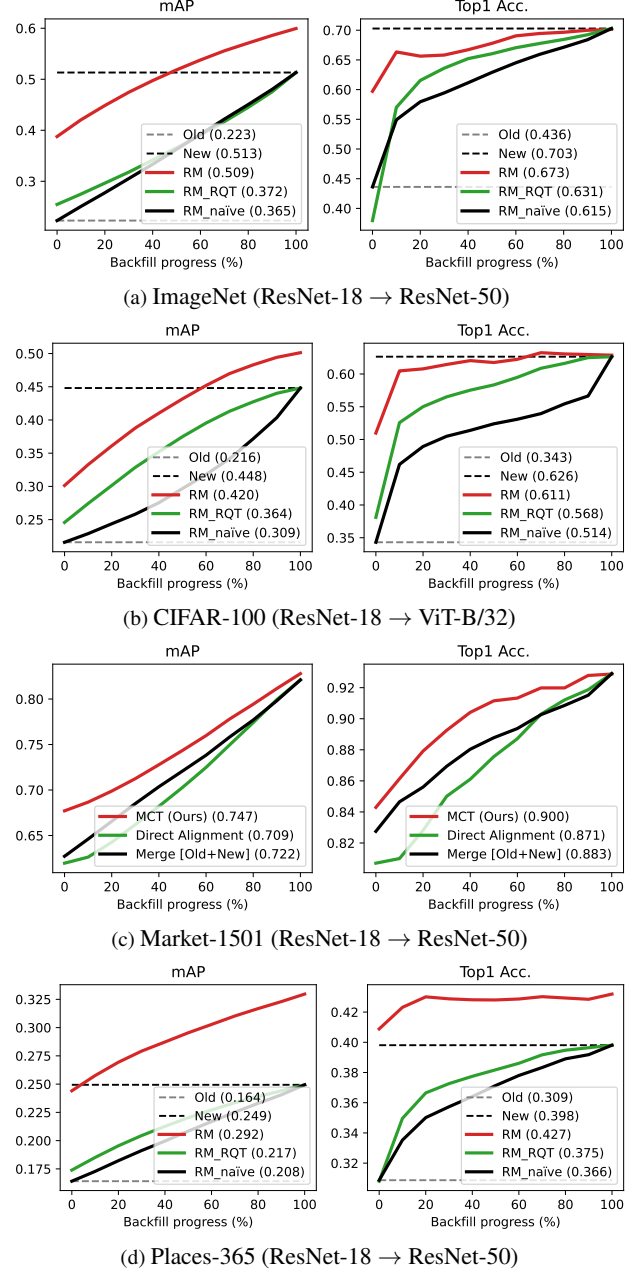


Figure 8. Experimental results with heterogeneous model upgrades. Our naïve rank merge between different architectures still achieves promising performance curves in various settings, and our full algorithm exhibits significantly better results.

sidering the distance metrics of both systems simultaneously helps to achieve better metric compatibility and consequently stronger merge results.

7. Conclusion

We presented a novel compatible training framework for effective and efficient online backfilling. We first addressed

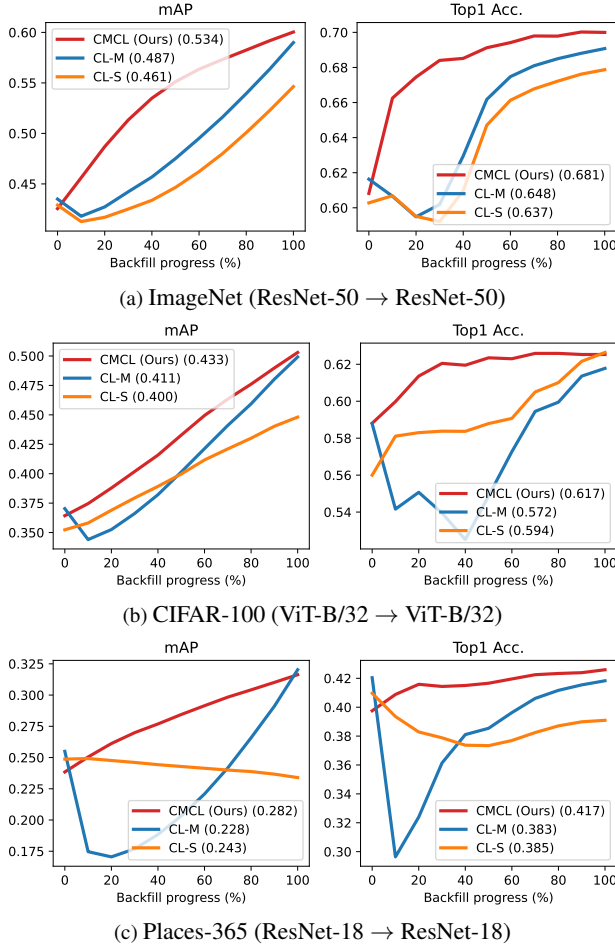


Figure 9. Ablation study of the cross-model contrastive learning loss on several datasets. CMCL outperforms other ablative models, CL-M and CL-S, which validates that the distance calibration plays a crucial role for effective rank merging.

the inherent trade-off between compatibility and discriminability, and proposed a practical alternative, online backfilling, to handle this dilemma. Our distance rank merge framework elegantly sidesteps this issue by bridging the gap between old and new models, and our metric-compatible learning further enhances the merge results with distance calibration. Our framework was validated via extensive experiments with significant improvement. We believe our work will provide a fundamental and practical foundation for promoting new directions in this line of research.

References

- [1] Yan Bai, Jile Jiao, Shengsen Wu, Yihang Lou, Jun Liu, Xue-tao Feng, and Ling-Yu Duan. Dual-tuning: Joint prototype transfer and structure regularization for compatible feature learning. *arXiv preprint arXiv:2108.02959*, 2021. 2
- [2] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICLR*, 2020. 2, 8
- [3] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021. 6
- [4] Rahul Duggal, Hao Zhou, Shuo Yang, Yuanjun Xiong, Wei Xia, Zhuowen Tu, and Stefano Soatto. Compatibility-aware heterogeneous visual search. In *CVPR*, 2021. 1
- [5] Albert Gordo, Jon Almazán, Jerome Revaud, and Diane Larlus. Deep image retrieval: Learning global representations for image search. In *ECCV*, 2016. 1
- [6] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *CVPR*, 2016. 6
- [7] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschiot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. *NeurIPS*, 2020. 5
- [8] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 7
- [9] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 6
- [10] Wei Li, Rui Zhao, Tong Xiao, and Xiaogang Wang. Deepreid: Deep filter pairing neural network for person re-identification. In *CVPR*, 2014. 1
- [11] Yixuan Li, Jason Yosinski, Jeff Clune, Hod Lipson, and John Hopcroft. Convergent learning: Do different neural networks learn the same representations? *arXiv preprint arXiv:1511.07543*, 2015. 2
- [12] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016. 7
- [13] Qiang Meng, Chixiang Zhang, Xiaoqiang Xu, and Feng Zhou. Learning compatible embeddings. In *ICCV*, 2021. 1, 2
- [14] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018. 5
- [15] Xiao Pan, Hao Luo, Weihua Chen, Fan Wang, Hao Li, Wei Jiang, Jianming Zhang, Jianyang Gu, and Peike Li. Dynamic gradient reactivation for backward compatible person re-identification. *arXiv preprint arXiv:2207.05658*, 2022. 2
- [16] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *NeurIPS*, 2019. 7
- [17] Vivek Ramanujan, Pavan Kumar Anasosalu Vasu, Ali Farhadi, Oncel Tuzel, and Hadi Pouransari. Forward compatible training for large-scale embedding retrieval systems. In *CVPR*, 2022. 2, 5, 6, 7, 8
- [18] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015. 4, 6

- [19] Yantao Shen, Yuanjun Xiong, Wei Xia, and Stefano Soatto. Towards backward-compatible representation learning. In *CVPR*, 2020. 1, 2, 3, 6, 7
- [20] Shupeng Su, Binjie Zhang, Yixiao Ge, Xuyuan Xu, Yexin Wang, Chun Yuan, and Ying Shan. Privacy-preserving model upgrades with bidirectional compatible training in image retrieval. *arXiv preprint arXiv:2204.13919*, 2022. 1, 2, 6, 7, 8
- [21] Yi Sun, Yuheng Chen, Xiaogang Wang, and Xiaoou Tang. Deep learning face representation by joint identification-verification. *NIPS*, 2014. 1
- [22] Timmy ST Wan, Jun-Cheng Chen, Tzer-Yi Wu, and Chu-Song Chen. Continual learning for visual search with backward consistent feature embedding. In *CVPR*, 2022. 1, 2
- [23] Fei Wang, Liren Chen, Cheng Li, Shiyao Huang, Yanjie Chen, Chen Qian, and Chen Change Loy. The devil of face recognition is in the noise. In *ECCV*, 2018. 1
- [24] Liwei Wang, Lunjia Hu, Jiayuan Gu, Zhiqiang Hu, Yue Wu, Kun He, and John Hopcroft. Towards understanding learning representations: To what extent do different neural networks learn the same representation. *NeurIPS*, 2018. 2
- [25] Binjie Zhang, Yixiao Ge, Yantao Shen, Yu Li, Chun Yuan, Xuyuan Xu, Yexin Wang, and Ying Shan. Hot-refresh model upgrades with regression-free compatible training in image retrieval. In *ICLR*, 2021. 2, 6
- [26] Binjie Zhang, Yixiao Ge, Yantao Shen, Shupeng Su, Chun Yuan, Xuyuan Xu, Yexin Wang, and Ying Shan. Towards universal backward-compatible representation learning. *arXiv preprint arXiv:2203.01583*, 2022. 2
- [27] Liang Zheng, Liyue Shen, Lu Tian, Shengjin Wang, Jingdong Wang, and Qi Tian. Scalable person re-identification: A benchmark. In *ICCV*, 2015. 6
- [28] Bolei Zhou, Agata Lapedriza, Jianxiong Xiao, Antonio Torralba, and Aude Oliva. Learning deep features for scene recognition using places database. *NIPS*, 2014. 4, 6