Unsupervised Lifelong Person Re-identification via Contrastive Rehearsal

Hao Chen[1,2], Benoit Lagadec[2], and Francois Bremond[1]

[1] Inria, Université Côte d'Azur, 06902 Valbonne, France
{hao.chen, francois.bremond}@inria.fr
[2] European Systems Integration, 06110 Le Cannet, France
benoit.lagadec@esifrance.net

Abstract. Existing unsupervised person re-identification (ReID) methods focus on adapting a model trained on a source domain to a fixed target domain. However, an adapted ReID model usually only works well on a certain target domain, but can hardly memorize the source domain knowledge and generalize to upcoming unseen data. In this paper, we propose unsupervised lifelong person ReID, which focuses on continuously conducting unsupervised domain adaptation on new domains without forgetting the knowledge learnt from old domains. To tackle unsupervised lifelong ReID, we conduct a contrastive rehearsal on a small number of stored old samples while sequentially adapting to new domains. We further set an image-to-image similarity constraint between old and new models to regularize the model updates in a way that suits old knowledge. We sequentially train our model on several large-scale datasets in an unsupervised manner and test it on all seen domains as well as several unseen domains to validate the generalizability of our method. Our proposed unsupervised lifelong method achieves strong generalizability, which significantly outperforms previous lifelong methods on both seen and unseen domains. Code will be made available at https://github.com/chenhao2345/UCR.

Keywords: Re-identification, lifelong learning, contrastive learning, knowledge accumulation

1 Introduction

Person re-identification (ReID) targets at matching a person of interest across non-overlapping cameras. Although significant improvement has been witnessed in both supervised [52,19] and unsupervised [15,6] person ReID, traditional methods only consider the performance of a single fixed target domain. In the single target domain scenario, people usually assume that all training data is available before training and deploying a ReID model. However, a real-world video monitoring system can record new data every day and from new locations, when new cameras are added into an existing system. How to adapt a model to new data without catastrophic forgetting on old knowledge has become a key point for training a generalizable and robust ReID model.

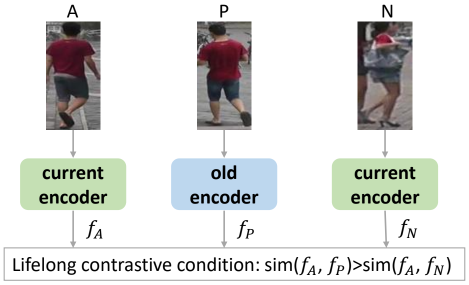
A
P
N
current encoder
old encoder
current encoder
f_A
f_P
f_N
Lifelong contrastive condition: sim(f_A, f_P)>sim(f_A, f_N)

Fig. 1: Unsupervised lifelong contrastive condition: the representation of an anchor (A) should always be more similar to a pseudo positive (P) than a pseudo negative (N), even though f_A and f_N are encoded with same current domain knowledge while f_P is encoded with old domain knowledge.

Towards a generalizable ReID model, *lifelong person ReID* [35,45] has been recently proposed to incrementally accumulate domain knowledge from several seen datasets. Lifelong person ReID is related to incremental (or continuous) learning [29], which aims at incrementally adding new classes or new domain knowledge into an existing model. As training and test sets in person ReID have non-overlapping identities, lifelong person ReID is defined as a domain-incremental learning task. A lifelong trained model has proven to be effective on every seen domain, as well as on unseen domains. However, previous lifelong person ReID relies on supervised cross-domain fine-tuning. When new data is recorded every day, people have to annotate new data manually before deployment, which is cumbersome and time-consuming. Replacing supervised cross-domain fine-tuning with unsupervised domain adaptation can maximally enhance the flexibility of a lifelong person ReID algorithm in real-world deployments.

In this paper, we propose a new *unsupervised lifelong person ReID* task to simultaneously explore 1) the possibility of training a generalizable model on incrementally added domains without human supervision and 2) the possibility of mitigating catastrophic forgetting problem neglected in traditional unsupervised person ReID. To train unsupervised lifelong person ReID, we have to consider learning unsupervised new domain representations and maintaining old domain knowledge simultaneously. In such context, we incorporate pseudo label based contrastive learning and rehearsal-based incremental learning into an *unsupervised contrastive rehearsal* (UCR) method, which tackles the forgetting problem during the unsupervised representation learning.

In our proposed UCR, a small number of old domain samples and their corresponding cluster prototypes are stored in long-term memory buffers. While adapting a model to a new domain, rehearsing stored old domain samples helps to prevent forgetting old knowledge. Given a frozen old domain model and a current domain model, we set a lifelong contrastive condition in Fig. 1: an old sample should always be closer to its pseudo positives than any pseudo negatives regardless of domain changes. Based on this condition, we try to retrieve positive pairs across different domain knowledge, which effectively mitigates forgetting in an unsupervised manner. Moreover, given a batch of old samples, the image-to-image similarity calculated by the old domain model and the new domain model should be consistent. We thus regularize the image-to-image representation relationship between old and new domain models, so that the new domain model can be updated in a way that suits old knowledge.

To summarize, our contributions are: 1) We propose a challenging but practical unsupervised lifelong person ReID task, which targets at incrementally learning a generalizable ReID model without human supervision. 2) We propose a contrastive rehearsal method and a representation relationship constraint to mitigate the forgetting problem in unsupervised lifelong person ReID. 3) Extensive experiments on both seen datasets and unseen datasets validate the effectiveness of our proposed method in unsupervised lifelong person ReID.

2 Related Work

Person ReID. Depending on the number of training/test domains and availability of human annotation, recent person ReID research is conducted under different settings. As the most studied setting, supervised person ReID [8,52,31,19] has shown impressive performance on large-scale datasets thanks to deep learning methods and human annotation. However, as a fine-grained retrieval task, a ReID model trained on one domain is hard to generalize to other domains. Unsupervised domain adaptation [14,15,11] and fully unsupervised ReID [43,6] are proposed to adjust a ReID model to a target domain with unlabeled target domain images. On the other hand, domain generalization ReID [40,24,10] is proposed to jointly train multiple labeled domains, in order to learn a generalizable model that can still work on unseen domains. But in most real-world cases, it is hard to prepare all training data in advance. Instead, new domain data can be recorded when time and season change or a new camera is installed. Supervised lifelong person ReID [35,45] is thus proposed to learn incrementally added new domains. However, continuously annotating new domains can be a cumbersome task for ReID system administrators. In this paper, we introduce unsupervised lifelong person ReID to maximally improve the flexibility of lifelong person ReID in the real-world deployments. We propose a contrastive rehearsal method to mitigate the catastrophic forgetting during the sequential unsupervised domain adaptation. Our proposed method is mainly related to contrastive learning and lifelong learning.

Contrastive learning. The main idea of contrastive learning is to maximize the representation similarity between a positive pair composed of differently augmented views of a same image, so that a model can understand the augmented variance is noise. While attracting a positive pair, some contrastive methods also consider other images as negatives and push away negatives stored in a memory bank [46,17] or in a large mini-batch [9]. Contrastive methods show great performance in unsupervised representation learning, which makes it a main approach in unsupervised person ReID. Based on clustering generated pseudo labels, state-of-the-art unsupervised person ReID methods build positive pairs with cluster centroids [15], camera-aware centroids [43], mini-batch hardest positives [6] and generated positive views [7]. However, all these methods are designed for single-domain unsupervised ReID, in which only representations from a single domain are contrasted. Differently, we propose to mitigate the catastrophic forgetting by contrasting representations across current and old domain knowledge.

Lifelong learning. Lifelong (also called incremental or continuous) learning aims at learning new classes or new domains without forgetting old knowledge. Several approaches have been proposed to address the forgetting problem in lifelong learning. One of the most intuitive approaches is rehearsal (also called recall or replay) learning , in which a small number of old data, such as raw images [36,3,4] or feature vectors [23], are stored to remind the new model of old knowledge. Another approach [48,5] consists in regularizing model updates on new data in a way that does not contradict the old knowledge. The third approach is mainly based on knowledge distillation [29,1,12], which considers the old model as a teacher for the new model. Two supervised lifelong person ReID methods have been recently proposed. In AKA [35], authors distill old knowledge via a learnable knowledge graph to the new domain model. GwFReID [45] stores 2 images per old domain identity for rehearsal and regularizes coherence between old domain and new domain models in both representation and classifier prediction levels. On the other hand, several attempts have been made in unsupervised lifelong adaptation, such as setting gradient regularization [41] in contrastive learning and consolidating the internal distribution [38]. However, general lifelong adaptation [41,38] has fixed classes across different domains, which are not suitable for lifelong ReID that has to learn fine-grained identity representations from totally different classes across domains.

3 Methodology

3.1 Overview

Given a stream of N person ReID datasets, unsupervised lifelong person ReID aims at learning a generalizable model via sequential unsupervised learning on the training set of each domain $D_1 \rightarrow D_2 \rightarrow ... \rightarrow D_N$. An unsupervised lifelong person ReID pipeline can be defined as one fully unsupervised ReID step on D_1 followed by several unsupervised domain adaptation steps on $D_2, ..., D_N$. After the whole pipeline, the adapted model is tested respectively on the test set of each seen domain, as well as on multiple unseen domains.

We present a new UCR method for lifelong person ReID. The general architecture of UCR is illustrated in Fig. 2. In the first step, we train a model on unlabeled images in the domain D_1 through a fully unsupervised ReID method. We follow state-of-the-art fully unsupervised ReID methods to use pseudo label based contrastive learning as a baseline, which considers both an online encoder and a momentum encoder, where the momentum encoder serves as a knowledge

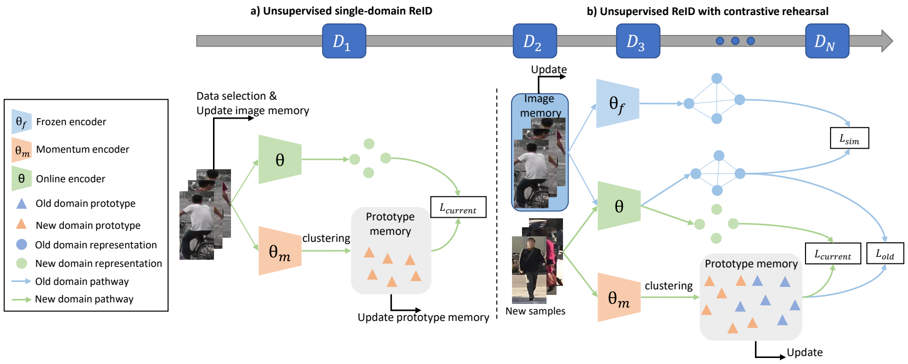

Fig. 2: General architecture of our proposed method. On the first domain D_1, we only follow the new domain pathway ($\rightarrow$) to conduct current domain contrastive learning $\mathcal{L}_{current}$. On the following domains $D_2 \rightarrow ... \rightarrow D_N$, we follow both new domain pathway ($\rightarrow$) and old domain rehearsal pathway($\rightarrow$) to conduct image-to-prototype contrastive learning $\mathcal{L}_{current}$ and $\mathcal{L}_{old}$, as well as similarity constraint $\mathcal{L}_{sim}$.

collector that gradually accumulates knowledge of each seen domain. To stabilize pseudo labels with momentum representations, the momentum encoder (weights noted as θ_m) is updated by exponential moving averaged weights of the online encoder (weights noted as θ):

$$\theta_m^t = \alpha\theta_m^{t-1} + (1 - \alpha)\theta^t \tag{1}$$

where the hyper-parameter α controls the update speed of the momentum encoder. t and $t - 1$ refer respectively to the current and last iteration. We extract all image representations with the stable momentum encoder and generate corresponding pseudo labels with a density-based clustering algorithm DBSCAN [13]. Based on the clustered pseudo labels, we build cluster prototypes for a current domain image-to-prototype contrastive loss $\mathcal{L}_{current}$ (described in Section 3.2) on the current domain D_1. To mitigate the forgetting, we build an image memory and a prototype memory to store old domain images and cluster prototypes for rehearsal. The memory buffers are updated after training each domain.

In the following adaptation step on domain D_i, we continue using the prototype contrastive loss $\mathcal{L}_{current}$ on new domain images to learn new knowledge, wherein pseudo labels are generated on the current domain images in the same way as the first step. On the other hand, we freeze the momentum encoder from the last step ($\theta_m^{D_{i-1}} \rightarrow \theta_f^{D_i}$) as an old knowledge expert model. Based on our lifelong contrastive condition that an old sample encoded by the new model θ should be close to its prototype, we formulate an old domain rehearsal loss $\mathcal{L}_{old}$ in Section 3.3. We further set an image-to-image similarity constraint (Section 3.4) to regularize the model updates during the continuous adaptation.

The overall unsupervised lifelong loss is defined as:

$$\mathcal{L}_{overall} = \mathcal{L}_{current} + \mathcal{L}_{old} + \lambda_{sim}\mathcal{L}_{sim} \tag{2}$$

3.2 Current domain contrastive baseline

Inside an unsupervised lifelong ReID pipeline, our model incrementally learns new knowledge on a current domain $D^c = \{(x_1^c, y_1^c), ..., (x_{N_{D_i}}^c, y_{N_{D_i}}^c)\}$ where N_{D_i} is the number of images and y is the clustered pseudo label of x. For a current domain image x_i^c, $f(x_i^c|\theta)$ and $f(x_i^c|\theta_m)$ denote respectively the online and the momentum representations. The prototype of a cluster a is defined as the averaged momentum representations of all the samples with a same pseudo label y_a:

$$p_a^c = \frac{1}{N_a} \sum_{x_i^c \in y_a} f(x_i^c|\theta_m) \tag{3}$$

When x_i^c belongs to the cluster a, a cluster prototype contrastive loss [15] can be defined as:

$$\mathcal{L}_{cluster} = \mathbb{E}[-\log \frac{\exp{(f(x_i^c|\theta) \cdot p_a^c/\tau_p)}}{\sum_{j=1}^{|P^c|} \exp{(f(x_i^c|\theta) \cdot p_j/\tau_p)}}] \tag{4}$$

where $|P^c|$ is the total number of clusters in the current domain and τ_p is a temperature hyper-parameter.

$\mathcal{L}_{cluster}$ makes samples in a cluster converge to a common prototype and get far away from other clusters. As a ReID dataset is usually recorded across different cameras, minimizing the intra-cluster variance from different camera style has proven to be effective in person ReID [43]. Supposing the current domain is recorded by N_C cameras $\mathcal{C} = \{c_1, ..., c_{N_C}\}$, an intra-cluster camera prototype is defined as the averaged momentum representation of all the samples with a same pseudo label y_a that are recorded from a same camera c_b:

$$p_{ab}^c = \frac{1}{N_{ab}} \sum_{x_i^c \in y_a \cap x_i^c \in c_b} f(x_i^c|\theta_m) \tag{5}$$

When x_i^c has a pseudo label y_a and is recorded from c_b, a camera prototype contrastive loss can be defined as:

$$\mathcal{L}_{cam} = \mathbb{E}[-\frac{1}{N_C} \sum_{j \in \mathcal{C}} \log \frac{\exp{(f(x_i^c|\theta) \cdot p_{aj}^c/\tau_c)}}{\sum_{k=1}^{N_{neg}+1} \exp{(f(x_i^c|\theta) \cdot p_k^c/\tau_c)}}] \tag{6}$$

where τ_c is a camera contrastive temperature hyper-parameter. N_{neg} hardest negative camera prototypes from the current domain are selected to enhance the model discriminability. $\mathcal{L}_{cam}$ maximizes the similarity between a representation and all the camera prototypes within a same cluster to reduce intra-cluster variance.

An overall loss on the current domain combines Eq. (4) and (6) with a balancing hyper-parameter λ_{cam}:

$$\mathcal{L}_{current} = \mathcal{L}_{cluster} + \lambda_{cam}\mathcal{L}_{cam} \tag{7}$$

Remark. By filtering out intra-cluster variance from different camera styles, $\mathcal{L}_{cam}$ purifies the current domain knowledge before being accumulated into the model. However, $\mathcal{L}_{cam}$ relies on camera labels, which make our method more ReID-specific. In fact, $\mathcal{L}_{cam}$ could be replaced with other techniques that do not require camera labels, such as contrasting mini-batch hardest positives [6] (see Supplementary Materials).

3.3 Old domain contrastive rehearsal

At the end of each step, we store all the cluster prototypes into a prototype memory and K_{mem} images per cluster into an image memory. To reduce pseudo label noise for contrastive rehearsal, for each cluster, we select K_{mem} images that have highest cosine similarity with the cluster prototype as reliable images to be stored. For the prototype memory, only general cluster prototypes but not camera-aware prototypes are stored to keep the memory buffer in a reasonable size.

At the beginning of adaptation on the domain D_i, the stored old domain cluster prototypes $P^o = \{P^{D_1}, ..., P^{D_{i-1}}\}$ are concatenated with current cluster prototypes $P^c = \{P^{D_i}\}$ (encoded by the current momentum encoder θ_m at the beginning of each epoch) to update the prototype memory $P = P^o \cup P^c$. Given an old sample x_i^o of identity y_a, if the old knowledge is well maintained in the current model, the online representation $f(x_i^o|\theta)$ encoded by the current domain online encoder θ should have the highest similarity score with the stored p_a^o encoded by the old domain encoder from the prototype memory. Thus, we construct an old domain contrastive rehearsal loss on stored old samples to remind the current model of old domain knowledge by maximizing the similarity between the old domain image representation $f(x_i^o|\theta)$ and the stored corresponding prototype p_a^o, while minimizing the similarity between $f(x_i^o|\theta)$ and other prototypes in the prototype memory:

$$\mathcal{L}_{old} = \mathbb{E}[-\log \frac{\exp\left(f(x_i^o|\theta) \cdot p_a^o/\tau_p\right)}{\sum_{j=1}^{|P|} \exp\left(f(x_i^o|\theta) \cdot p_j/\tau_p\right)}] \tag{8}$$

where $|P|$ is the number of cluster prototypes in the prototype memory and τ_p is the prototype temperature hyper-parameter same as Eq. (4). As a cluster prototype (the averaged representation of all cluster samples) contains generic information of a cluster, the prototype memory enables the current model to have access to generic old domain cluster information without storing all the images.

3.4 Image-to-image Similarity Constraint

Technically, person ReID is a representation similarity ranking problem, in which the objective is to have high similarity scores between positive pairs and low similarity scores between negative pairs. However, when a model is adapted into a new domain, the similarity relationship between old domain samples could be affected by the new domain knowledge. As the similarity relationship between same images should be consistent before and after a domain adaptation step, we propose an image-to-image similarity constraint loss that regularizes the similarity relationship updates in a way that does not contradict the old knowledge. As the frozen old model θ_f from the last domain can be regarded as an expert on the old domain, the similarity relationship calculated by the frozen model can be regarded as a reference for regularizing the current model θ updates.

Given a mini-batch of old images $\{x_1^o, ..., x_{N_{bs}}^o\}$ where N_{bs} is the batch size, the image-to-image similarity distribution can be calculated with a softmax function on the cosine similarity between each image pair in the mini-batch. The image-to-image similarity between two old images x_i^o and x_j^o is calculated with both the online encoder θ and the momentum encoder θ_m:

$$P_{i,j} = \frac{\exp\left(< f(x_i^o|\theta) \cdot f(x_j^o|\theta_m) > /\tau_s\right)}{\sum_{k=1}^{N_{bs}} \exp\left(< f(x_i^o|\theta) \cdot f(x_k^o|\theta_m) > /\tau_s\right)} \tag{9}$$

where $< \cdot >$ denotes the normalized cosine similarity and τ_s is a similarity temperature hyper-parameter.

For the same mini-batch, we calculate the image-to-image similarity distribution with the frozen old model θ_f as a reference for the constraint. The reference similarity between two old domain images x_i^o and x_j^o is:

$$Q_{i,j} = \frac{\exp\left(< f(x_i^o|\theta_f) \cdot f(x_j^o|\theta_f) > /\tau_s\right)}{\sum_{k=1}^{N_{bs}} \exp\left(< f(x_i^o|\theta_f) \cdot f(x_k^o|\theta_f) > /\tau_s\right)} \tag{10}$$

We formulate an image-to-image similarity constraint loss with a Kullback-Leibler (KL) Divergence between the two distributions:

$$\mathcal{L}_{sim} = \mathcal{D}_{KL}(P||Q) \tag{11}$$

By minimizing $\mathcal{L}_{sim}$, we encourage the similarity relationship P calculated with current domain knowledge to be consistent with that calculated with old domain knowledge Q.

Remark. Here, the similarity $P_{i,j}$ is calculated in online/momentum (θ/θ_m) format, which is the similarity between current online representations and accumulated momentum representations. Such online/momentum similarity encourages the online encoder θ updates in a way that is consistent with the accumulated momentum encoder θ_m, which is better than only consider online/online (θ/θ) similarity.

4 Experiment

4.1 Datasets and Evaluation Protocols

As DukeMTMC-reID dataset [37] has been taken down from the website, we do not follow previous lifelong ReID benchmarks [45,35]. Instead, we set up a new lifelong person ReID benchmark, which contains 3 seen datasets for domain-incremental training and 9 unseen datasets for generalizability evaluation, as

Type	Dataset	#img	#train id	#train img	#test id
Seen	Market [50]	36036	751	12936	750
	Cuhk-Sysu [47]	23435	5532	15088	2900
	MSMT17 [44]	124068	1041	32621	3060
Unseen	VIPeR [16]	1264	-	-	316
	PRID [21]	1134	-	-	649
	GRID [30]	1275	-	-	125
	iLIDS [51]	476	-	-	60
	CUHK01 [27]	1942	-	-	486
	CUHK02 [26]	7264	-	-	239
	SenseReID [49]	4428	-	-	1718
	CUHK03 [28]	14097	-	-	100
	3DPeS [2]	1012	-	-	96

Table 1: Dataset statistics. Unseen domains are only used for testing.

shown in Table 1. Compared with previous supervised lifelong ReID benchmarks [45,35] with DukeMTMC-reID, our benchmark contains less seen domains but more unseen domains, which can better evaluate the model generalizability.

Seen datasets. We use three large-scale datasets as seen domains, including Market, Cuhk-Sysu and MSMT17. We use the ReID version of Cuhk-Sysu dataset, in which each person bounding box is cropped from raw images using the ground-truth annotation. We formulate a 3-step domain-incremental training in the order Market→Cuhk-Sysu→MSMT17. Our model is trained sequentially on the training set of each seen dataset and is tested on the test set of each dataset after the final step. Cumulative Matching Characteristics (CMC) at Rank1 accuracy and mean Average Precision (mAP) are used in our experiments.

Unseen dataset. We use 9 person ReID datasets to maximally evaluate the model generalizability on different unseen domains, including VIPeR, PRID, GRID, iLIDS, CUHK01, CUHK02, SenseReID, CUHK03 and 3DPeS. These 9 datasets cover all the unseen domains that are considered in previous supervised lifelong ReID methods [45,35] and domain generalizable ReID methods [40]. We use the traditional training/test split on CUHK03 dataset. Rank1 accuracy and mAP results are respectively reported on the test set of each unseen domain after the final step.

4.2 Implementation details

Training. Our method is implemented under Pytorch [34] framework. The total training time with 4 Nvidia 1080Ti GPUs is around 6 hours. We use an ImageNet [39] pre-trained ResNet50 [18] as our backbone network. We resize all images to 256×128 and augment images with random horizontal flipping, cropping, Gaussian blurring and erasing [54]. At each step (domain), we train our framework 30 epochs with 400 iterations per epoch using a Adam [25] optimizer with a weight decay rate of 0.0005. The learning rate is set to 0.00035 with a warm-up scheme

Training order: Market→Cuhk-Sysu→MSMT17										
Method	Memory (image per id)	Type	Market		Cuhk-Sysu		MSMT17		Average	
			mAP	Rank1	mAP	Rank1	mAP	Rank1	mAP	Rank1
SpCL [15]	0	U	18.8	39.2	70.9	74.4	10.5	24.8	33.4	46.1
ICE [6]	0	U	29.0	60.4	72.5	76.3	21.8	49.0	41.1	61.9
BL+LwF [29]	0	UL	34.3	67.7	73.3	76.8	23.5	52.5	43.7	65.6
BL+SPD [42]	0	UL	33.4	65.3	74.9	78.1	27.5	58.9	45.3	67.4
BL+iCaRL [36]	2	UL	38.6	67.7	80.6	83.0	26.3	56.3	48.5	69.0
BL+Co^2L [4]	2	UL	43.5	72.7	78.5	81.0	30.4	61.5	50.8	71.7
BL+**UCR**	2	UL	57.6	83.0	83.2	85.6	25.4	54.1	**55.4**	**74.3**
AKA [35]	0	SL	57.7	78.6	77.0	80.0	9.2	21.5	48.0	60.0
BL(GT)+LwF [29]	0	SL	39.6	68.9	80.7	83.7	44.1	72.0	54.8	74.9
BL(GT)+SPD [42]	0	SL	37.7	66.5	80.8	83.0	41.5	69.7	53.3	73.1
BL(GT)+iCaRL [36]	2	SL	32.8	60.6	85.4	87.7	43.5	70.8	53.9	73.1
BL(GT)+Co^2L [4]	2	SL	48.6	74.5	84.1	86.3	45.2	72.3	59.3	77.7
BL(GT)+**UCR**	2	SL	59.3	82.7	88.3	90.0	40.8	67.5	**62.8**	**80.1**

Table 2: Seen-domain results (%) of unsupervised single-domain (U), supervised lifelong (SL) and unsupervised lifelong (UL) methods. 'BL' denotes our current domain baseline. 'BL(GT)' refers to replacing pseudo labels with ground truth labels.

in the first 10 epochs. No learning rate decay is used in the training. Pseudo labels on the current domain are updated on re-ranked Jaccard distance [53] at the beginning of each epoch with a DBSCAN [13], in which the minimum cluster sample number is set to 4 and the distance threshold is set to 0.55. The momentum encoder is updated with a momentum hyper-parameter $\alpha = 0.999$. Following [6], we set $\tau_p = 0.5$, $\tau_c = 0.07$, $N_{neg} = 50$ and $\lambda_{cam} = 0.5$ in Eq. (6) in the baseline. We use a grid search on τ_s and λ_{sim}, which are presented in Section 4.3. After the whole training, only the momentum encoder is saved for inference. We provide more details of our algorithm in Supplementary Materials.

Mini-batch composition. To balance the model ability on old domains and a current domain, we separately take a mini-batch of current domain images and a mini-batch of old domain images of a same batch size, which is set to 32 in our experiments. Furthermore, we use a random identity sampler to construct mini-batches to handle the imbalanced images of different identities. Following the clustering setting on the current domain (each cluster has at least 4 neighbors), the 32 current domain images are composed of 8 identities and 4 images per identity. Following the supervised lifelong ReID method [45], we set $K_{mem} = 2$ to store 2 images per cluster. The 32 old domain images are thus composed of 16 identities and 2 images per identity.

Compared methods. We re-implement three types of methods for comparison on lifelong person ReID, including unsupervised single-domain methods, supervised lifelong methods and unsupervised lifelong methods.

The unsupervised single-domain methods include SpCL [15] and ICE [6], which are trained sequentially on each seen domain.

The lifelong methods include four general-purpose lifelong methods (LwF [29], SPD [42], iCaRL [36] and Co^2L [4]) and two ReID-specific lifelong methods

Training order: Market→Cuhk-Sysu→MSMT17																						
Method	Memory (per id)	Type	VIPeR		PRID		GRID		iLIDS		CUHK01		CUHK02		SenseReID		CUHK03		3DPeS		Average	
			mAP	R1	mAP	R1	mAP	R1	mAP	R1	mAP	R1	mAP	R1	mAP	R1	mAP	R1	mAP	R1	mAP	R1
SpCL [15]	0	U	31.7	22.8	9.2	4.0	10.6	5.6	58.7	48.3	45.2	44.9	40.2	37.9	28.1	22.1	8.1	22.2	35.5	43.1	29.7	27.9
ICE [6]	0	U	35.7	25.9	39.0	29.0	20.6	14.4	71.4	61.7	60.6	60.0	48.2	45.8	33.6	27.9	17.3	29.7	48.5	55.4	41.7	38.9
BL+LwF [29]	0	UL	41.2	32.3	49.0	37.0	31.9	23.2	76.8	66.7	63.4	62.0	54.1	52.3	37.6	31.2	21.7	35.0	53.1	59.4	47.7	44.3
BL+SPD [42]	0	UL	41.3	30.1	29.7	22.0	27.4	19.2	79.2	71.7	67.5	67.1	55.4	55.9	39.7	33.1	21.0	37.9	50.8	57.9	45.8	43.9
BL+iCaRL [36]	2	UL	45.9	35.1	48.6	39.0	32.5	22.4	78.9	71.7	66.2	66.9	57.5	55.6	43.8	36.4	24.1	41.6	56.7	64.4	50.5	48.1
BL+Co^2L [4]	2	UL	47.7	37.0	51.1	40.0	28.4	20.0	80.6	73.3	70.5	71.0	62.3	60.5	44.0	36.5	29.8	43.9	60.0	66.8	52.7	49.9
BL+**UCR**	2	UL	47.7	37.0	55.5	47.0	40.6	31.2	85.3	81.7	69.8	68.8	68.0	65.3	47.0	39.5	33.0	48.0	64.9	71.3	**56.8**	**54.4**
AKA [35]	0	SL	37.9	28.8	31.0	21.0	24.0	15.2	70.6	60.0	54.1	53.3	47.2	43.9	34.8	28.1	19.5	19.3	43.9	56.3	40.3	36.2
GwFReID* [45]	2	SL	-	43.2	-	-	-	-	-	69.5	-	-	-	-	-	-	-	40.2	-	64.9	-	-
BL(GT)+LwF [29]	0	SL	51.5	40.5	41.8	33.0	26.8	20.8	79.1	71.7	74.4	75.8	62.2	61.7	44.0	37.4	29.8	47.4	55.3	60.9	51.7	49.9
BL(GT)+SPD [42]	0	SL	48.7	37.7	25.5	15.0	23.6	16.8	81.9	75.0	70.9	71.8	60.9	60.5	44.2	36.8	26.9	45.7	54.6	65.8	48.6	47.2
BL(GT)+iCaRL [36]	2	SL	52.4	41.5	45.9	37.0	32.4	22.4	82.0	75.0	69.1	70.3	62.5	61.1	47.0	39.7	33.3	51.0	57.2	63.4	53.5	51.3
BL(GT)+Co^2L [4]	2	SL	56.3	46.2	52.3	41.0	28.4	20.8	83.5	76.7	77.5	78.3	67.0	65.1	48.9	41.2	37.1	54.5	60.5	67.8	56.8	54.6
BL(GT)+**UCR**	2	SL	57.7	47.5	56.0	44.0	40.6	31.2	87.9	85.0	75.0	75.8	72.9	72.4	52.9	45.2	39.8	57.0	66.5	75.7	**61.0**	**59.3**

Table 3: Unseen-domain results (%) of unsupervised single-domain (U), supervised lifelong (SL) and unsupervised lifelong (UL) methods. 'BL' denotes our current domain baseline. 'BL(GT)' refers to replacing pseudo labels with ground truth labels. As we do not have source code for re-implementation, GwFReID* is reported on Market→Duke→Cuhk-Sysu→MSMT17, which benefit from more domain data than our setting.

(AKA [35] and GwFReID [45]). LwF and SPD are pure distillation-based methods, which do not store old samples for rehearsal. LwF uses a prediction-level cross-entropy distillation [20] between old and new domain models. SPD distills mid-level feature similarity between old and new domain models. iCaRL and Co^2L are rehearsal-based methods. iCaRL conducts prediction-level distillation on new and stored old images for rehearsal. Co^2L proposes an asymmetric supervised contrastive loss and a relation distillation for supervised continual learning. For general-purpose methods LwF, SPD, iCaRL and Co^2L, we **combine the same current-domain contrastive baseline (Section 3.2) and the lifelong learning techniques of each paper** to convert these methods to person ReID and conduct a fair comparison with our method. For example, in LwF, we combine our contrastive baseline for learning current domain knowledge and the prediction-level distillation for mitigating the forgetting.

Seen-domain non-forgetting evaluation. We report seen-domain results after the final step in Table 2. Designed for maximally learning domain-specific features inside a single domain, SpCL and ICE can not learn domain-agnostic generalized features for lifelong ReID. Among lifelong methods, the rehearsal-based methods iCaRL and Co^2L yield better averaged performance than the pure distillation-based methods LwF and SPD. Under the unsupervised lifelong setting, our proposed UCR outperforms the second best method Co^2L by 4.6% on averaged mAP and 2.6% on averaged Rank1. We also replace the pseudo labels with ground truth labels to compare in the supervised lifelong setting. Our re-implementation of LwF, SPD, iCaRL and Co^2L outperform the ReID-specific method AKA. Under the supervised lifelong setting, our proposed UCR outperform the second best method Co^2L by 3.5% on averaged mAP and 2.4% on averaged Rank1. We further draw first seen domain (Market) mAP/Rank1

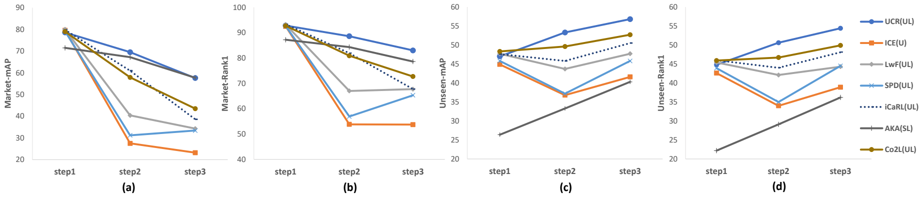

Fig. 3: Non-forgetting evaluation on the first seen domain: (a) mAP and (b) Rank1 on Market-1501. Generalizability evaluation: (c) averaged mAP and (d) averaged Rank1 on the unseen domains.

variation curves after each step in Fig. 3 (a) and (b), which confirm that our UCR has a slower forgetting rate.

Unseen-domain generalizability evaluation. To compare the generalizability of each method, we report unseen-domain results in Table 3. Similar to seen-domain results, SpCL and ICE can hardly learn domain-agnostic generalized features, which leads to low performance on unseen domains. On the contrary, lifelong methods accumulate knowledge from each domain and eventually learn domain-agnostic generalized features. With the same baseline, the rehearsal-based methods iCaRL and Co^{2}L outperform the pure distillation-based methods LwF and SPD. Under the unsupervised lifelong setting, our proposed UCR outperforms the second best method Co^{2}L by 4.1% on averaged mAP and 4.5% on averaged Rank1. We also compare the supervised performance of AKA, GwFReID, LwF, SPD, iCaRL and Co^{2}L on the unseen domains. Since the source code of GwFReID is not available, we report the results from the original paper, which benefit from one more seen domain (DukeMTMC) but are still lower than our results on iLIDS, CUHK03 and 3DPeS. Our proposed UCR outperforms the second best method Co^{2}L by 4.2% on averaged mAP and 4.7% on averaged Rank1. We draw averaged mAP/Rank1 variation curves on all the unseen domains after each step in Fig. 3 (c) and (d), which show that our UCR achieves better generalizability than other methods.

More analysis. Our method shares certain similarity with Co^{2}L, because both methods use contrastive losses for rehearsal. However, Co^{2}L only uses current domain samples as anchors, while old domain samples are served as negatives in the contrastive loss. Our method UCR uses both current and old domain samples as anchors to retrieve the corresponding cluster prototypes from a prototype memory. In addition, we set a constraint to regularize the similarity relationship update, which is effective on similarity ranking problem person ReID.

4.3 Ablation study

The performance improvement of UCR over the baseline mainly comes from our proposed old domain contrastive rehearsal and the image-to-image similarity

Method	Seen-Avg		Unseen-Avg	
	mAP	Rank1	mAP	Rank1
Baseline	45.7	67.4	47.3	45.1
$+\mathcal{L}_{old}$	49.4	69.2	49.6	48.1
$+\mathcal{L}_{sim}$	53.7	73.6	54.7	51.3
$+\mathcal{L}_{old}+\mathcal{L}_{sim}$	**55.4**	**74.3**	**56.8**	**54.4**

Table 4: Ablation study on the contrastive rehearsal loss $\mathcal{L}_{old}$ and the similarity constraint loss $\mathcal{L}_{sim}$. We report the averaged results.

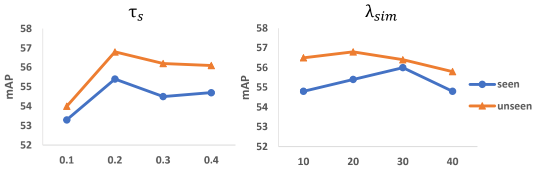

Fig. 4: Sensitivity to similarity constraint temperature τ_s and balancing coefficient λ_{sim}. We report the averaged results on seen and unseen domains.

K_{mem}	Seen-Avg		Unseen-Avg	
	mAP	R1	mAP	R1
1	53.0	72.7	55.2	52.7
2	55.4	74.3	56.8	54.4
4	57.1	75.2	58.0	55.2
8	58.1	75.8	**58.7**	55.6
All	**59.3**	**75.9**	58.3	**55.9**

Table 5: Number of images per pseudo identity in the memory.

K_{mem}	Seen-Avg		Unseen-Avg	
	mAP	R1	mAP	R1
Nearest	**55.4**	**74.3**	**56.8**	**54.4**
Farthest	55.1	74.2	55.5	53.4
Random	55.0	73.7	55.3	53.6

Table 6: memory sample selection based on the distance between images and the cluster prototype.

Method	Seen-Avg		Unseen-Avg	
	mAP	R1	mAP	R1
Co^2L(UL)[4]	53.7	66.1	50.5	46.7
UCR(UL)	**55.3**	**71.6**	**52.1**	**49.6**
AKA(SL) [35]	43.9	58.9	41.1	38.1
Co^2L(SL)[4]	59.6	69.0	52.4	49.7
UCR(SL)	**59.9**	**72.7**	**57.3**	**56.1**

Table 7: Second training order: MSMT17 → CuhkSysu → Market.

constraint. To validate the effectiveness of each component, we conduct ablation experiments by gradually adding one of them to the baseline. In Table 4, 'Baseline' refers to sequential training each new domain with $\mathcal{L}_{current}$ in Eq. (7) without any non-forgetting techniques. The performance on both seen and unseen domains can be improved by adding the old domain contrastive rehearsal '$+\mathcal{L}_{old}$'. The overall performance boost from the similarity constraint '$+\mathcal{L}_{sim}$' is more significant, which indicates that regularizing image-to-image relation is more effective than regularizing image-to-prototype relation for unsupervised lifelong ReID. One possible explanation is that prototypes are built by clustering pseudo labels, which can be noisy for lifelong ReID. '$+\mathcal{L}_{old}+\mathcal{L}_{sim}$' denotes our full UCR method. We conclude that the two components are always beneficial and complementary for both seen and unseen domains.

4.4 Parameter analysis

We analyze the sensitivity of our method to important hyper-parameters on lifelong person ReID performance. We first vary the number of stored images per identity K_{mem} to study the sensitivity of our memory size. As shown in Table 5, storing more samples with a larger K_{mem} results in less forgetting and better generalizability. In our experiments, we set $K_{mem} = 2$ to fairly compare with GwFReID [45]. In real-world deployments, users can choose a smaller K_{mem} to save storage space and get slightly lower results, or a larger K_{mem} to get better results. We further vary values of hyper-parameters τ_s and λ_{sim} for the similarity constraint. Based on the results in Fig. 4, we set the similarity temperature $\tau_s = 0.2$ and the balancing weight $\lambda_{sim} = 20$. We analyze clustering parameters and compare backbone networks in Supplementary Materials.

4.5 More discussion

Data selection for memory update. To update the image memory after each step, for each cluster, we can select K_{mem} samples nearest to the cluster prototype, or farthest to the cluster prototype, or random samples. As shown in Table 6, storing nearest samples for rehearsal achieves slightly better performance, because nearest samples are more reliable clustered samples that bring in less pseudo label noise.

Training order. Our primary training order follows previous lifelong methods [45], which starts from a medium domain Market and ends with the largest domain MSMT17. However, it is hard to control the order of upcoming domains in the real world. We test a second order MSMT17→Cuhk-Sysu→Market, which makes it easier to forget more knowledge on the largest domain MSMT17. As shown in Table 7, UCR still significantly outperforms state-of-the-art methods AKA and Co^2L. Please refer to Supplementary Materials for more details.

Memory size. We use both image and prototype memory buffers in our rehearsal-based method UCR. With imperfect clustering pseudo labels and $K_{mem} = 2$, our image memory stores approximately 640(Market)$\times$2+1050(Cuhk-Sysu)$\times$2+1900(MSMT17)$\times 2 = 7180$ images $\approx$ 11.8% of all the training images (Market, Cuhk-Sysu and MSMT17). On the other hand, our prototype memory stores approximately 640(Market)+1050(Cuhk-Sysu)+1900(MSMT17)= 3590 prototype vectors (dimension $1 \times 2048 \times 1 \times 1$)$\approx$29.4 MB, which is negligible compared with storing dataset images (for example, MSMT17$\approx$2.5 GB).

Limitation and future work. Even though we only store 2 images and 1 prototype representation vector per identity, the memory size can still grow rapidly due to daily recorded new data in real-world deployments. An interesting direction for our future work is to explore how to reduce the number of stored images. For example, setting up a metric to select more representative images and filter out less representative images from each domain can be a possible solution.

5 Conclusion

In this paper, we introduce a challenging but practical task, namely unsupervised lifelong person ReID, to explore the possibility of learning a generalizable model via sequential unsupervised adaptation on incrementally added domains. To tackle the catastrophic forgetting after the domain adaptation, we propose an unsupervised contrastive rehearsal (UCR) method, which rehearses a small number of old samples in a contrastive manner. We also set a similarity relationship constraint to regularize the model update in a way that suits old knowledge. In comparison with previous lifelong methods, our proposed UCR achieves better non-forgetting performance on seen domains and better generalizability on unseen domains.

References

1. Aljundi, R., Babiloni, F., Elhoseiny, M., Rohrbach, M., Tuytelaars, T.: Memory aware synapses: Learning what (not) to forget. In: ECCV (2018)
2. Baltieri, D., Vezzani, R., Cucchiara, R.: 3dpes: 3d people dataset for surveillance and forensics. In: Proceedings of the 2011 joint ACM workshop on Human gesture and behavior understanding (2011)
3. Castro, F.M., Marín-Jiménez, M.J., Guil, N., Schmid, C., Alahari, K.: End-to-end incremental learning. In: ECCV (2018)
4. Cha, H., Lee, J., Shin, J.: Co2l: Contrastive continual learning. In: ICCV (2021)
5. Chaudhry, A., Dokania, P.K., Ajanthan, T., Torr, P.H.: Riemannian walk for incremental learning: Understanding forgetting and intransigence. In: ECCV (2018)
6. Chen, H., Lagadec, B., Bremond, F.: Ice: Inter-instance contrastive encoding for unsupervised person re-identification. In: ICCV (2021)
7. Chen, H., Wang, Y., Lagadec, B., Dantcheva, A., Bremond, F.: Joint generative and contrastive learning for unsupervised person re-identification. In: CVPR (2021)
8. Chen, T., Ding, S., Xie, J., Yuan, Y., Chen, W., Yang, Y., Ren, Z., Wang, Z.: Abd-net: Attentive but diverse person re-identification. In: ICCV (2019)
9. Chen, T., Kornblith, S., Norouzi, M., Hinton, G.: A simple framework for contrastive learning of visual representations. In: ICML (2020)
10. Dai, Y., Li, X., Liu, J., Tong, Z., Duan, L.Y.: Generalizable person re-identification with relevance-aware mixture of experts. In: CVPR (2021)
11. Dai, Y., Liu, J., Sun, Y., Tong, Z., Zhang, C., Duan, L.Y.: Idm: An intermediate domain module for domain adaptive person re-id. In: ICCV (2021)
12. Douillard, A., Chen, Y., Dapogny, A., Cord, M.: Plop: Learning without forgetting for continual semantic segmentation. In: CVPR (2021)
13. Ester, M., Kriegel, H.P., Sander, J., Xu, X.: A density-based algorithm for discovering clusters in large spatial databases with noise. In: KDD (1996)
14. Ge, Y., Chen, D., Li, H.: Mutual mean-teaching: Pseudo label refinery for unsupervised domain adaptation on person re-identification. In: ICLR (2020)
15. Ge, Y., Zhu, F., Chen, D., Zhao, R., Li, H.: Self-paced contrastive learning with hybrid memory for domain adaptive object re-id. In: NeurIPS (2020)
16. Gray, D., Tao, H.: Viewpoint invariant pedestrian recognition with an ensemble of localized features. In: ECCV (2008)
17. He, K., Fan, H., Wu, Y., Xie, S., Girshick, R.: Momentum contrast for unsupervised visual representation learning. In: CVPR (2020)
18. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR (2016)
19. He, S., Luo, H., Wang, P., Wang, F., Li, H., Jiang, W.: Transreid: Transformer-based object re-identification. In: ICCV (2021)
20. Hinton, G., Vinyals, O., Dean, J.: Distilling the knowledge in a neural network. arXiv preprint arXiv:1503.02531 (2015)
21. Hirzer, M., Beleznai, C., Roth, P.M., Bischof, H.: Person Re-Identification by Descriptive and Discriminative Classification. In: Proc. Scandinavian Conference on Image Analysis (SCIA) (2011)
22. Ioffe, S., Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift. In: ICML (2015)
23. Iscen, A., Zhang, J., Lazebnik, S., Schmid, C.: Memory-efficient incremental learning through feature adaptation. In: ECCV (2020)

24. Jin, X., Lan, C., Zeng, W., Chen, Z., Zhang, L.: Style normalization and restitution for generalizable person re-identification. In: CVPR (2020)
25. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: ICLR (2015)
26. Li, W., Wang, X.: Locally aligned feature transforms across views. CVPR (2013)
27. Li, W., Zhao, R., Wang, X.: Human reidentification with transferred metric learning. In: ACCV (2012)
28. Li, W., Zhao, R., Xiao, T., Wang, X.: Deepreid: Deep filter pairing neural network for person re-identification. CVPR (2014)
29. Li, Z., Hoiem, D.: Learning without forgetting. IEEE TPAMI (2018)
30. Loy, C.C., Xiang, T., Gong, S.: Multi-camera activity correlation analysis. In: CVPR (2009)
31. Luo, H., Gu, Y., Liao, X., Lai, S., Jiang, W.: Bag of tricks and a strong baseline for deep person re-identification. In: CVPR Workshops (2019)
32. van der Maaten, L., Hinton, G.: Visualizing data using t-sne. JMLR (2008)
33. Pan, X., Luo, P., Shi, J., Tang, X.: Two at once: Enhancing learning and generalization capacities via ibn-net. In: ECCV (2018)
34. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., Chintala, S.: Pytorch: An imperative style, high-performance deep learning library. In: NeurIPS (2019)
35. Pu, N., Chen, W., Liu, Y., Bakker, E.M., Lew, M.S.: Lifelong person re-identification via adaptive knowledge accumulation. In: CVPR (2021)
36. Rebuffi, S.A., Kolesnikov, A., Sperl, G., Lampert, C.H.: icarl: Incremental classifier and representation learning. CVPR (2017)
37. Ristani, E., Solera, F., Zou, R., Cucchiara, R., Tomasi, C.: Performance measures and a data set for multi-target, multi-camera tracking. In: ECCV workshops (2016)
38. Rostami, M.: Lifelong domain adaptation via consolidated internal distribution. NeurIPS (2021)
39. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A., Fei-Fei, L.: Imagenet large scale visual recognition challenge. IJCV (2015)
40. Song, J., Yang, Y., Song, Y.Z., Xiang, T., Hospedales, T.M.: Generalizable person re-identification by domain-invariant mapping network. CVPR (2019)
41. Tang, S., Su, P., Chen, D., Ouyang, W.: Gradient regularized contrastive learning for continual domain adaptation. In: AAAI (2021)
42. Tung, F., Mori, G.: Similarity-preserving knowledge distillation. ICCV (2019)
43. Wang, M., Lai, B., Huang, J., Gong, X., Hua, X.S.: Camera-aware proxies for unsupervised person re-identification. In: AAAI (2021)
44. Wei, L., Zhang, S., Gao, W., Tian, Q.: Person transfer gan to bridge domain gap for person re-identification. In: CVPR (2018)
45. Wu, G., Gong, S.: Generalising without forgetting for lifelong person re-identification. In: AAAI (2021)
46. Wu, Z., Xiong, Y., Yu, S.X., Lin, D.: Unsupervised feature learning via non-parametric instance discrimination. In: CVPR (2018)
47. Xiao, T., Li, S., Wang, B., Lin, L., Wang, X.: Joint detection and identification feature learning for person search. CVPR (2017)
48. Yoon, J., Yang, E., Lee, J., Hwang, S.J.: Lifelong learning with dynamically expandable networks. In: ICLR (2018)

49. Zhao, H., Tian, M., Sun, S., Shao, J., Yan, J., Yi, S., Wang, X., Tang, X.: Spindle net: Person re-identification with human body region guided feature decomposition and fusion. CVPR (2017)
50. Zheng, L., Shen, L., Tian, L., Wang, S., Wang, J., Tian, Q.: Scalable person re-identification: A benchmark. ICCV (2015)
51. Zheng, W.S., Gong, S., Xiang, T.: Associating groups of people. In: BMVC (2009)
52. Zheng, Z., Yang, X., Yu, Z., Zheng, L., Yang, Y., Kautz, J.: Joint discriminative and generative learning for person re-identification. In: CVPR (2019)
53. Zhong, Z., Zheng, L., Cao, D., Li, S.: Re-ranking person re-identification with k-reciprocal encoding. In: CVPR (2017)
54. Zhong, Z., Zheng, L., Kang, G., Li, S., Yang, Y.: Random erasing data augmentation. In: AAAI (2020)

Supplementary Materials

In the supplementary material, we provide more details about our algorithm in Section A, the possible alternatives for our current domain contrastive baseline in Section B and the performance of second training order in Section C. We proceed to analyze the generalizability of our method after each training step with a domain gap visualization in Section D. We further analyze the sensitivity to clustering hyper-parameters in Section E and the performance with a stronger backbone network in Section F.

A Algorithm Details

Our proposed UCR is composed of one fully unsupervised learning step on the first domain D_1 and several unsupervised incremental learning steps on the following domains $D_2, ..., D_N$. In this way, we incrementally accumulate knowledge from each seen domain into a same momentum encoder θ_m. To help readers better understand our proposed method, we present algorithm details in Algorithm 1.

B Current domain contrastive baseline

For unsupervised lifelong person ReID, purifying the intra-cluster variance in each current domain helps to mitigate the noise in the knowledge accumulation.

In our main paper, the current domain baseline is defined as $\mathcal{L}_{current} = \mathcal{L}_{cluster} + \lambda_{cam}\mathcal{L}_{cam}$, where $\mathcal{L}_{cam}$ uses camera labels to purify the camera style variance in current domain knowledge. Actually, $\mathcal{L}_{cam}$ can be replaced by other variance reduction techniques, such as hard instance contrastive loss [6]. The hard instance contrastive loss $\mathcal{L}_{hard}$ maximizes the similarity between an anchor representation $f(x_i^c|\theta)$ (superscript c denotes the current domain) and the hardest pseudo-positive $f(x_{hard}^c|\theta_m)$ in the mini-batch:

$$\mathcal{L}_{hard} = \mathbb{E}[-\log \frac{\exp{(< f(x_i^c|\theta) \cdot f(x_{hard}^c|\theta_m) >)}}{\sum_{j=1}^{N_{neg}+1} \exp{(< f(x_i^c|\theta) \cdot f(x_j^c|\theta_m) >)}}] \tag{12}$$

where $< \cdot >$ denotes the normalized cosine similarity. $f(x_{hard}^c|\theta_m)$ is the pseudo-positive that has the lowest cosine similarity with $f(x_i^c|\theta)$. N_{neg} is the number of pseudo-negatives in the mini-batch. $\mathcal{L}_{hard}$ reduces the distance between anchor and hard samples to encourage the compactness of a cluster, so that intra-cluster variance can be reduced in the current domain.

We compare three possible baselines for learning current domain knowledge: 1) only contrasting general cluster prototypes $\mathcal{L}_{current} = \mathcal{L}_{cluster}$, 2) reducing current domain noise by mining hard positives $\mathcal{L}_{current} = \mathcal{L}_{cluster} + \mathcal{L}_{hard}$ and 3)

Algorithm 1: UCR for unsupervised lifelong person ReID.

```
Input: Unlabeled seen domains $D_1, D_2, ..., D_N$.
for $domain = D_1$ to $D_N$ do
    if First domain $D_1$ then
        # Single-domain unsupervised learning #
        for $epoch = 1$ to $E_{max}$ do
            Generate pseudo labels on $D_1$;
            Calculate current domain prototypes $P^c$ in Eq. (3) and camera
              prototypes in Eq. (5) on $D_1$;
            Initialize prototype memory $P = P^c$;
            for $iter = 1$ to $I_{max}$ do
                Sample a mini-batch from current domain $D_1$ for $\mathcal{L}_{current}$ in
                  Eq. (7);
                Update $\theta$ with $\mathcal{L}_{current}$;
                Update $\theta_m$ with Eq. (1);
            end
        end
        Initialize image memory with $K_{mem}$ images per identity;
    else
        # Unsupervised lifelong learning with rehearsal #
        for $epoch = 1$ to $E_{max}$ do
            Generate pseudo labels on $D_i$;
            Calculate current domain prototypes $P^c$ in Eq. (3) and camera
              prototypes in Eq. (5) on $D_i$;
            Update prototype memory $P = P^o \cup P^c$;
            for $iter = 1$ to $I_{max}$ do
                Sample a mini-batch from current domain $D_i$ for $\mathcal{L}_{current}$ in
                  Eq. (7);
                Sample a mini-batch from image memory for $\mathcal{L}_{old}$ in Eq. (8)
                  and $\mathcal{L}_{sim}$ in Eq. (11);
                Train $\theta$ with $\mathcal{L}_{overall}$ in Eq. (2);
                Update $\theta_m$ with Eq. (1);
            end
        end
        Update image memory with $K_{mem}$ images per identity;
    end
    $\theta_f \leftarrow \theta_m$ and $\theta \leftarrow \theta_m$;
end
return $\theta_m$ after the final domain $D_N$
```

reducing current domain noise with camera labels $\mathcal{L}_{current} = \mathcal{L}_{cluster} + \mathcal{L}_{cam}$. As shown in Table 8, the cluster prototypes in the case 1 can be noisy if we do not reduce intra-cluster variance, leading to unsatisfactory rehearsal effectiveness of $\mathcal{L}_{old}$. Reducing the distance between hard samples inside a cluster with $\mathcal{L}_{hard}$ (case 2) can mitigate the noise in cluster prototypes. The most effective way is to leverage camera labels with $\mathcal{L}_{cam}$ (case 3), as presented in the main paper.

Baseline	Method	Seen-Avg		Unseen-Avg	
		mAP	Rank1	mAP	Rank1
1) $\mathcal{L}_{current} = \mathcal{L}_{cluster}$	$\mathcal{L}_{current}$	37.9	55.7	38.8	37.3
	$+\mathcal{L}_{old}$	38.7	54.7	34.7	33.5
	$+\mathcal{L}_{sim}$	43.9	62.6	44.0	42.8
	$+\mathcal{L}_{old}+\mathcal{L}_{sim}$	44.9	62.3	42.9	42.0
2) $\mathcal{L}_{current} = \mathcal{L}_{cluster} + \mathcal{L}_{hard}$	$\mathcal{L}_{current}$	38.8	58.0	44.5	41.9
	$+\mathcal{L}_{old}$	47.0	65.2	48.4	46.5
	$+\mathcal{L}_{sim}$	53.0	67.2	54.3	50.7
	$+\mathcal{L}_{old}+\mathcal{L}_{sim}$	52.6	68.7	55.1	52.7
3) $\mathcal{L}_{current} = \mathcal{L}_{cluster} + \mathcal{L}_{cam}$ (used in main paper)	$\mathcal{L}_{current}$	45.7	67.4	47.3	45.1
	$+\mathcal{L}_{old}$	49.4	69.2	49.6	48.1
	$+\mathcal{L}_{sim}$	53.7	73.6	54.7	51.3
	$+\mathcal{L}_{old}+\mathcal{L}_{sim}$	**55.4**	**74.3**	**56.8**	**54.4**

Table 8: Ablation study on the old domain contrastive rehearsal loss $\mathcal{L}_{old}$ and the similarity distillation loss $\mathcal{L}_{sim}$ on the alternative baselines. We report the averaged results on seen and unseen domains.

C Second training order

To complement Table 7 in the main paper, we provide more details about the performance on each dataset under the second training order MSMT17→Cuhk-Sysu→Market. The second training order starts from the largest dataset MSMT17 and ends by a medium dataset Market, which is opposite to our primary training order Market→Cuhk-Sysu→MSMT17. As shown in Table 9 and Table 10, our method outperforms state-of-the-art methods AKA and Co^{2}L on both seen and unseen domains by a clear margin. Our UCR(UL) yields better non-forgetting performance on the first domain MSMT17 than UCR(SL), because the clustering in UCR(UL) generates approximately 1050 pseudo-identitites for Cuhk-Sysu, while UCR(SL) contains 5532 ground truth identitites for Cuhk-Sysu. UCR(SL) stores more cluster prototypes and images for Cuhk-Sysu, which decreases the weight of MSMT17 in the memory buffers.

D Domain gap visualization

We use t-SNE [32] visualization on 200 randomly selected samples from each unseen domain to roughly estimate the domain gap encoded in representations after

Second training order: MSMT17→Cuhk-Sysu→Market										
Method	Memory (image per id)	Type	MSMT17		Cuhk-Sysu		Market		Average	
			mAP	Rank1	mAP	Rank1	mAP	Rank1	mAP	Rank1
BL+Co^2L [4]	2	UL	9.6	27.2	77.3	79.6	74.4	91.4	53.7	66.1
BL+**UCR**	2	UL	15.7	41.3	81.1	83.7	69.2	89.9	**55.3**	**71.6**
AKA [35]	0	SL	13.4	31.6	74.5	77.9	43.8	67.1	43.9	58.9
BL(GT)+Co^2L [4]	2	SL	9.9	26.4	83.9	85.7	84.9	94.8	59.6	69.0
BL(GT)+**UCR**	2	SL	12.7	36.3	87.9	89.5	79.0	92.2	**59.9**	**72.7**

Table 9: Results (%) of supervised lifelong methods (SL) and unsupervised lifelong methods (UL) on seen domains. 'BL' denotes our current domain baseline. 'BL(GT)' refers to replacing pseudo labels with ground truth labels.

Second training order: MSMT17→Cuhk-Sysu→Market																					
Method	Memory (per id)	Type	VIPeR		PRID		GRID		iLIDS		CUHK01		CUHK02		SenseReID		CUHK03		3DPeS		Average	
			mAP	R1	mAP	R1	mAP	R1	mAP	R1	mAP	R1	mAP	R1	mAP	R1	mAP	R1	mAP	R1	mAP	R1
BL+Co^2L [4]	2	UL	43.7	33.5	43.0	32.0	40.0	31.2	76.9	68.3	61.9	61.9	61.2	57.1	43.3	35.7	26.7	36.7	57.5	63.4	50.5	46.7
BL+**UCR**	2	UL	43.4	34.8	50.7	40.0	32.2	24.0	80.6	73.3	66.5	66.5	62.8	60.3	45.8	38.9	26.9	41.2	60.3	67.3	**52.1**	**49.6**
AKA [35]	0	SL	36.3	26.3	38.0	29.0	18.7	13.6	69.6	60.0	60.9	61.7	53.9	54.2	30.2	24.6	19.2	19.5	43.1	54.0	41.1	38.1
BL(GT)+Co^2L [4]	2	SL	46.6	37.7	44.5	34.0	39.6	30.4	77.2	68.3	64.6	63.6	64.5	63.0	48.4	40.3	30.1	44.7	55.9	65.8	52.4	49.7
BL(GT)+**UCR**	2	SL	51.3	41.8	50.3	37.0	43.1	33.6	85.6	80.0	73.9	73.4	73.2	73.6	49.5	41.2	28.7	53.2	59.8	70.8	**57.3**	**56.1**

Table 10: Results (%) of supervised lifelong methods (SL) and unsupervised lifelong methods (UL) on unseen domains. 'BL' denotes our current domain baseline. 'BL(GT)' refers to replacing pseudo labels with ground truth labels.

each training step. As shown in Figure 5, the domain gap is obvious before training, especially on GRID (green), iLIDS (red), PRID (orange) and 3DPeS (olive). The first step transfers our model from ImageNet distribution into Market person ReID distribution, which generally reduces the domain gap. The second step accumulates more domain knowledge into our model, making PRID (orange) and 3DPeS (olive) get closer to other domains. As MSMT17 contains more illumination and scenario diversity, the third step further reduces the domain gap, for instance, between GRID (green) and other domains. GRID (green) is recorded in underground stations that have an illumination level and backgrounds significantly different to other street camera recorded datasets. We conclude that our unsupervised lifelong method UCR effectively reduces the domain gap encoded in representations and incrementally learns domain-agnostic features.

E Clustering distance threshold

The distance threshold of DBSCAN is the maximal distance between two samples for one to be considered as in the neighborhood of the other. A larger distance threshold enlarges the radius of a cluster, making more samples be considered into a same cluster.

Our proposed method UCR conducts image-to-prototype contrastive learning with clustering generated pseudo labels, which can be affected by the clustering distance threshold. In our main paper, we follow ICE [6] to set the threshold to 0.55. As it is hard to decide a uniform threshold for upcoming datasets of different sizes, we evaluate the sensitivity of UCR to the clustering distance threshold. We can observe from Table 11 that different threshold values only bring slight

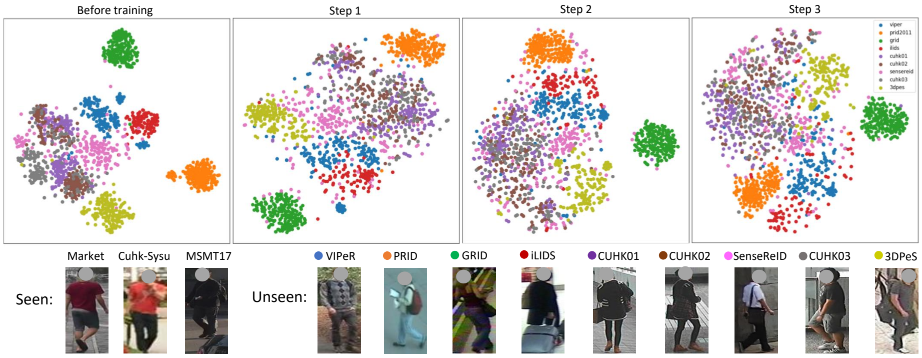

Fig. 5: T-SNE visualization of unseen-domain representations after each step.

seen-domain and unseen-domain performance variation, showing that UCR is robust to the clustering threshold.

Threshold	Seen-Avg		Unseen-Avg	
	mAP	R1	mAP	R1
0.45	55.0	73.4	56.6	54.3
0.5	55.3	74.0	**57.2**	**54.7**
0.55	55.4	74.3	56.8	54.4
0.6	**56.4**	**74.9**	56.4	54.1
0.65	55.4	73.4	55.6	52.1

Table 11: DBSCAN clustering distance threshold.

F Backbone Network

Our proposed unsupervised lifelong method UCR leverages multiple domains to learn generalized features that can achieve balanced performance on all the domains. The generalizability is strongly related to the backbone network. A ResNet50 backbone is used in the experiments of our main paper to have a fair comparison with previous methods. In fact, the performance of UCR can be further improved with a backbone with stronger generalizability, such as IBN-ResNet50 [33]. In IBN-ResNet50, authors propose to replace batch normalization [22] in ResNet with instance-batch normalization to enhance model generalizability. We compare the performance of using ResNet50 and IBN-ResNet50 as our backbone network in Table 12. IBN-ResNet50 significantly fills the performance gap between unsupervised and supervised lifelong settings. Under unsupervised lifelong setting, IBN-ResNet50 outperforms ResNet50 by a large margin. Under supervised lifelong setting, IBN-ResNet50 outperforms ResNet50 on unseen domains but not on seen domains, which indicates that instance-batch normalization brings in better generalizability rather than non-forgetting capacity when there is no label noise.

Backbone	Type	Seen-Avg		Unseen-Avg	
		mAP	R1	mAP	R1
ResNet50	UL	55.4	74.3	56.8	54.4
IBN-ResNet50	UL	**59.4**	**77.5**	**61.1**	**58.5**
ResNet50+GT	SL	**62.8**	80.1	61.0	59.3
IBN-ResNet50+GT	SL	61.9	**80.2**	**62.7**	**60.9**

Table 12: Comparison of backbone networks in our proposed UCR under unsupervised lifelong (UL) and supervised lifelong (SL) settings. 'GT' refers to ground truth.