Patch-based Knowledge Distillation for Lifelong Person Re-Identification

Zhicheng Sun
Peking University, Beijing 100871, P.R. China
sunzc@pku.edu.cn

Yadong Mu*
Peking University, Beijing 100871, P.R. China
myd@pku.edu.cn

ABSTRACT

The task of lifelong person re-identification aims to match a person across multiple cameras given continuous data streams. Similar to other lifelong learning tasks, it severely suffers from the so-called *catastrophic forgetting* problem, which refers to the notable performance degradation on previously-seen data after adapting the model to some newly incoming data. To alleviate it, a few existing methods have utilized knowledge distillation to enforce consistency between the original and adapted models. However, the effectiveness of such a strategy can be largely reduced facing the data distribution discrepancy between seen and new data. The hallmark of our work is using adaptively-chosen patches (rather than whole images as in other works) to pilot the forgetting-resistant distillation. Specifically, the technical contributions of our patch-based new solution are two-fold: first, a novel patch sampler is proposed. It is fully differentiable and trained to select a diverse set of image patches that stay crucial and discriminative under streaming data. Secondly, with those patches we curate a novel knowledge distillation framework. Valuable patch-level knowledge within individual patch features and mutual relations is well preserved by the two newly introduced distillation modules, further mitigating catastrophic forgetting. Extensive experiments on twelve person re-identification datasets clearly validate the superiority of our method over state-of-the-art competitors by large performance margins.

CCS CONCEPTS

• **Computing methodologies → Object identification**; **Lifelong machine learning**.

KEYWORDS

person re-identification, lifelong learning, knowledge distillation, patch selection

ACM Reference Format:
Zhicheng Sun and Yadong Mu. 2022. Patch-based Knowledge Distillation for Lifelong Person Re-Identification. In *Proceedings of the 30th ACM International Conference on Multimedia (MM '22), October 10–14, 2022, Lisboa, Portugal.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3503161.3548179

*Corresponding author.

MM '22, October 10–14, 2022, Lisboa, Portugal
© 2022 Association for Computing Machinery.
ACM ISBN 978-1-4503-9203-7/22/10…$15.00
https://doi.org/10.1145/3503161.3548179

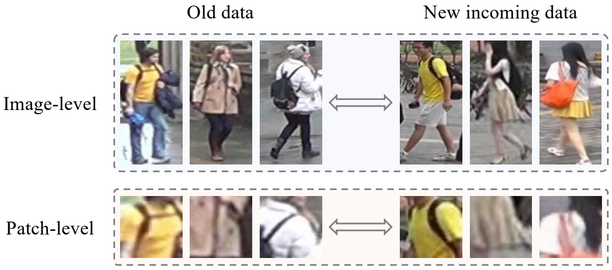


Figure 1: Illustration of the distribution gap between different person re-identification samples. As shown in this figure, our main observation in this work is: the distribution gap between images from old data and new incoming data partly stems from some image-level factors (for example different background clutters or camera viewing angle). It can be hopefully mitigated when the model operates in the level of most related and discriminative patches across the data streams.

1 INTRODUCTION

Person re-identification (ReID) aims to match pedestrian images captured from non-overlapping cameras. It has made remarkable progress in recent years thanks to the development of deep learning models and large-scale datasets [9, 18, 32, 57]. However, most of ReID methods assume that the training data can be accessed all at once, which limits their application to real-world streaming data, *e.g.*, millions of images produced every day by surveillance video systems. To investigate efficient and scalable learning algorithms for continuously incoming ReID data, the task of lifelong person ReID has recently been proposed [46, 56, 63, 69]. It requires the ReID model to simultaneously incorporate new information while preserving already learned knowledge.

Like other lifelong learning tasks [7, 47, 54, 62], lifelong person ReID faces a key challenge of catastrophic forgetting, *i.e.*, performance degradation on old datasets after updating the model on new ones. A straightforward solution is to keep records of old data and re-train the model periodically, but it is not always feasible due to storage limits and privacy concerns. In order to alleviate catastrophic forgetting without accessing old data, most lifelong ReID methods [46, 56, 69] have adopted a knowledge distillation strategy. At each stage, the current model distills knowledge from the old one by mimicking its behavior on new data. However, this method relies heavily on the relatedness between the old data and the new data, and may not perform well under dramatic distribution shift [2, 34]. Such distribution shift is common in lifelong person ReID due to large distribution gap caused by camera views, background variations etc. This issue was still inadequately studied in previous work, resulting in low effectiveness of knowledge distillation and non-negligible forgetting.

Bridging the distribution gap between old and new data has been extensively studied by unsupervised domain-adaptive ReID [12, 66, 75]. Patch-based feature learning [66] is a representative approach, which provides an inspiring observation that the gap between similar patches are smaller compared to similar images. As demonstrated in Figure 1, the selected patches are less affected by factors such as background clutters, therefore more robust under distribution shift than the holistic image. We are thus motivated to make full use of patch-level information for more effective knowledge distillation, thereby mitigating catastrophic forgetting. The main difficulty with this idea lies in two challenges: selecting patches that are similar to the previous distributions without accessing old data, and preserving crucial patch-level knowledge.

In this work, we propose a patch-based knowledge distillation framework for lifelong person ReID. First, a differentiable patch sampler is designed to select patch features by jointly optimizing two objectives: being close to the old data and carrying diverse information. The former objective can be effectively addressed based on the model's prediction confidence as in previous practice [19, 29, 35]. We also propose to achieve the latter via a multi-branch design along with a diversity loss. The selected patches are then fed to a patch classifier to distill knowledge from the old model. Secondly, we develop patch relation distillation to retain useful relational knowledge among patches. Considering the structure inherent in patch relations, namely that inter-instance relations outnumber informative intra-instance relations, we propose to treat these two types of patch relations in a separate manner. By enforcing consistency of intra-instance and inter-instance feature distances, both local correlations and global identity information are preserved.

The contributions of our paper are summarized as follows:

- A patch-based knowledge distillation framework is curated for lifelong person ReID. The proposed model exploits a key observation that the distribution gap induced by most related and discriminative image patches is significantly smaller than operating in an image level. This helps alleviate the so-called catastrophic forgetting in the interested task.
- A differentiable patch sampler is designed to minimize patch-level distribution gap with the guidance from model confidences. The sampled patches are used for knowledge distillation and make it more robust under the distribution shift.
- A patch relation distillation module is proposed to preserve relational knowledge among patch features. It works by distilling on both intra-instance and inter-instance feature distances.

2 RELATED WORKS

This section briefly surveys a few research thrusts that are tightly related to the research in this paper.

Person re-identification. Person ReID has made remarkable progress under various settings such as supervised learning [18, 40, 57], unsupervised learning [9, 36] and unsupervised domain adaptation [12, 15, 66, 75, 76]. However, these settings all assume that the training data can be access all at once and therefore struggle to generalize to continuously increasing real-world data. To tackle this problem, some methods for lifelong person ReID have recently been proposed [46, 56, 63, 69]. Sugianto *et al.* [56] apply the learning without forgetting method [34] to alleviate catastrophic forgetting. Wu *et al.* [63] characterize lifelong person ReID by unseen class recognition, domain generalization and class imbalanced learning, and then propose a comprehensive learning objective to address these problems. Zhao *et al.* [69] propose neighbor selection and consistency relaxation strategies for better scalability and generalization ability. Pu *et al.* [46] maintain a learnable knowledge graph to accumulate previous knowledge and generalize to unseen domains. However, the above methods distill knowledge at image-level, which may be disturbed by distribution shift, resulting in performance degradation on old data. We propose to distill knowledge at patch-level, which is more robust under distribution shift.

There have been a series of studies on patch-based ReID [32, 52, 66] and part-based ReID [55, 57, 70–72]. They either address the spatial misalignment problem between two person images [32, 52, 55, 70–72] or aim to learn discriminative local features [57, 66]. In contrast, we perform patch-based knowledge distillation to mitigate catastrophic forgetting in lifelong person ReID.

Lifelong learning. Lifelong learning methods have been developed in three major streams: expansion-based, rehearsal-based and regularization-based. Expansion-based methods [11, 41, 51] handle increasing knowledge by expanding model architecture on demand. Rehearsal-based methods alleviate catastrophic forgetting by recalling on stored [6, 38, 47] or synthesized [53, 62, 68] images of previous tasks. However, they impose strict requirements on storage space or image generator capacity. Regularization-based methods [1, 14, 28, 34] add regularization terms to limit the change of model parameters. The widely used learning without forgetting method [34] falls into this category, which preserves old knowledge with a distillation loss calculated on the new data.

Knowledge distillation. Knowledge distillation is first presented by Hinton *et al.* [21] and then widely used to transfer knowledge from one network to another network. Existing knowledge distillation methods can be summarized into three categories: logit distillation [3, 21] which matches the final predictions, feature distillation [49, 67] which matches the intermediate representations, and relation distillation [37, 42, 45, 60] which matches inter-sample relations. Li *et al.* [33] and Kim *et al.* [26] extend logit distillation and relation distillation to patch-level, but their methods generate patch features from a regular grid without any selection, bringing the dilemma of high computational cost or loss of fine-grained information. We achieve more efficient patch-based knowledge distillation with the proposed differentiable patch sampler.

Patch selection. Several prior works employ differentiable patch selection before downstream processing. Yang *et al.* [66] exploit the spatial transformer [23] to locate discriminative patches for feature extraction, but the training of the spatial transformer could be difficult [10]. Katharopoulos *et al.* [25] utilize learnable attention to sample informative patches from high resolution images. Tseng *et al.* [59] and Cordonnier *et al.* [10] formulate patch sampling as a top-K problem and solve it with the Gumbel-max trick[24, 65] and perturbed optimizers [5] respectively. However, these sampling-based methods draw samples from the same distribution, which may produce mutually similar patches conveying overlapping information. We propose to sample each patch from a different distribution and introduce an additional diversity loss to ensure diversity among selected patches.

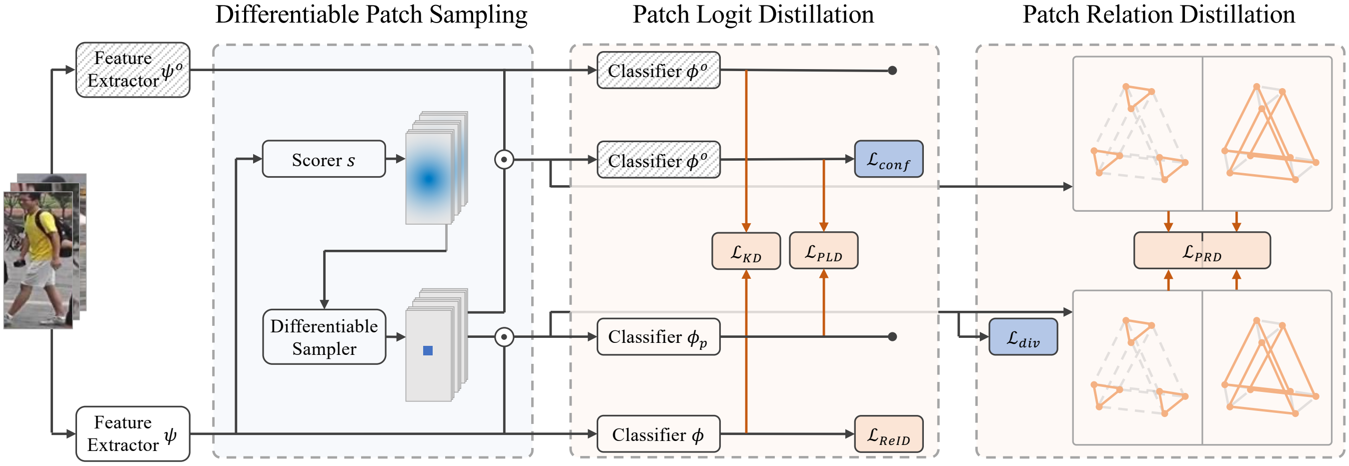


Figure 2: Overall architecture of our proposed framework. The loss functions marked with blue guide the patch sampler to minimize patch-level distribution gap. The loss functions marked with light orange enforce the ReID model to learn from new samples without catastrophic forgetting. Textured background represents that the module is frozen during training. ⊙ denotes element-wise product.

3 PROBLEM FORMULATION

Lifelong person ReID concerns the learning over streaming data, which we simulate with a stream of datasets $\mathcal{D} = \{D^1, \ldots, D^T\}$. Each dataset D^t consists of person images X^t and identity labels Y^t. During time t, the ReID model is optimized on the dataset D^t, aiming to learn from new samples without interfering with performance on previous datasets $D^1, \ldots, D^{t-1}$. Following the recent lifelong person ReID literature [46, 56, 69], we assume that the model cannot access training samples from previous steps, either directly or through an external memory. At the test time, the model is evaluated on the test split of all seen datasets and new unseen datasets for its non-forgetting performance and generalization ability, respectively.

4 METHOD

We propose a patch-based knowledge distillation framework for lifelong person ReID. As illustrated in Figure 2, it builds on an image-based base model (Section 4.1) and incorporates three new modules: a differentiable patch sampler (Section 4.2) for selecting a diverse set of patches that tend to be invariably important over streaming data and are used for piloting the distillation on the new data, patch logit distillation (Section 4.3) that encourages the current model to mimic the old one's prediction on the selected patches, and patch relation distillation (Section 4.4) that helps the model to retain various type of patch-level relational knowledge.

4.1 Base Model

We begin by introducing a base model that only uses image-level feature to address the lifelong person ReID problem. The base model f is a mapping from input images to logits (class scores over person identities). It can be broken down into three stages as $f = \phi \circ g \circ \psi$. First, the backbone network $\psi(\cdot)$ extracts a feature map from the input image, followed by the global average pooling $g(\cdot)$ that applies average pooling over the spatial dimension to generate a ReID feature from the feature map. Finally, the classifier head $\phi(\cdot)$ predicts logits from the ReID feature, which can be converted to probabilities with an activation function σ such as softmax. Furthermore, let a superscript o be the mark for something related to the old model. For example, $f^o = \phi^o \circ g \circ \psi^o$ denotes the model from the last time step, which is frozen during training.

The primary goal of the base model is to extract discriminative ReID-oriented features based on the current identity labels Y^t. Following the common practice of conventional ReID methods [18, 40], we introduce the ReID loss consisting of a cross-entropy loss that matches predicted probabilities to class labels and a triplet loss [20] that encourages intra-class compactness and inter-class separability. Given a mini-batch of samples $\{(x_i, y_i)\}_{i=1}^B$ from the current dataset D^t, where B is the mini-batch size, the ReID loss is defined as

$$\begin{aligned}\mathcal{L}_{ReID} =& \frac{1}{B}\sum_{i=1}^{B}\mathcal{L}_{CE}\left(y_i, \sigma(f(x_i))\right) \\ &+ \frac{1}{B}\sum_{i=1}^{B}\mathcal{L}_{tri}\left(g(\psi(x_i)), g(\psi(x_i^p)), g(\psi(x_i^n))\right),\end{aligned} \tag{1}$$

where x_i^p and x_i^n are the positive and negative samples for x_i respectively. All samples are from D^t only unless otherwise notified.

However, optimizing the model only by ReID loss may lead to catastrophic forgetting on previously seen datasets. To preserve learned knowledge, we enforce consistency between the current model and the old model using a distillation loss that minimizes the Kullback-Leibler divergence between the logits of the two models:

$$\mathcal{L}_{KD} = \frac{1}{B}\sum_{i=1}^{B}\mathrm{KL}\left(\sigma(f^o(x_i)/\tau) \,\middle\|\, \sigma(f(x_i)/\tau)\right), \tag{2}$$

where τ is a hyperparameter referred to as the temperature by Hinton *et al.* [21]. Then, the loss function for the base model is treated as a weighted sum of $\mathcal{L}_{ReID}$ and $\mathcal{L}_{KD}$:

$$\mathcal{L}_{base} = \mathcal{L}_{ReID} + \gamma_0 \mathcal{L}_{KD}, \tag{3}$$

where γ_0 is a trade-off factor between the ReID feature learning and knowledge distillation.

Despite the wide application of knowledge distillation in lifelong learning tasks [54, 56], it is known that this approach relies heavily on the relatedness of old and new data [2, 34] because distribution shift between these data can result in a gradual error build-up to the previous datasets [13]. Unfortunately, such distribution shift is common across ReID datasets, leading to poor non-forgetting performance. In order to mitigate the interference of distribution shift on knowledge distillation, we'll present our patch-based model in the following sections.

4.2 Differentiable Patch Sampling

In this section, we first explain the underlying mechanism of the proposed differentiable patch sampler in a simplified scenario, *i.e.*, sampling a single patch, and then introduce three key designs to guide the patch sampler to select a diverse set of patches that are less affected by the distribution shift.

Sampling a single patch. For the sake of computational efficiency, we sample the patch from a relatively small feature map instead of the original image. Specifically, given an input images x_i, we use the last feature maps $\psi(x_i), \psi^o(x_i)$ and treat them both as a set of candidate features. To model the sampling probability of each candidate feature, a learnable scorer is employed to predict a score vector s_i, where each value is subsequently converted to the sampling probability of the j-th candidate feature $\psi(x_i)_j$ (j indexes all possible neurons or image patches) using an activation function σ. Hence, the distribution of the sampled patch feature p_i is modeled as

$$P\left(p_i = \psi(x_i)_j\right) = \sigma(s_i)_j. \tag{4}$$

Directly choosing a maximum from the above distribution is non-differentiable. We adopt a differentiable alternative by using the Gumbel-max trick [24] to draw a discrete sample M_i:

$$M_i = \texttt{one\_hot}\left(\arg\max_j (s_i + G)_j\right), \tag{5}$$

where G is a vector of i.i.d. Gumbel noise samples, and employing a straight-through estimator [4, 24] to enable differentiation of arg max in the backward pass. Here, M_i determines the location of the sampled patch, so it is used as a shared mask to extract patch features of the two models from feature maps $\psi(x_i), \psi^o(x_i)$:

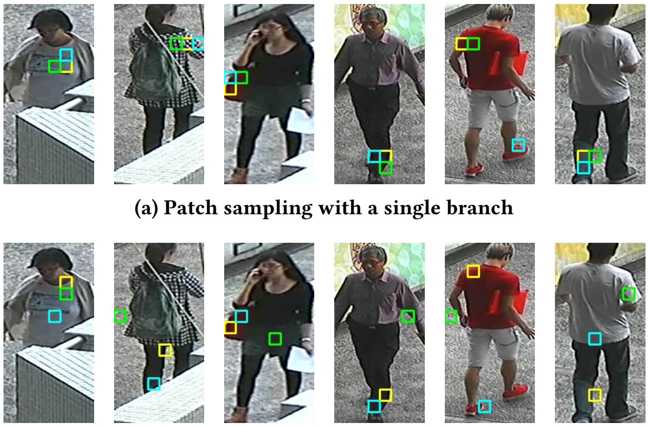
$$p_i = M_i \odot \psi(x_i), \quad p_i^o = M_i \odot \psi^o(x_i), \tag{6}$$

where $\odot$ denotes element-wise product. In this way, the patch sampling process becomes differentiable.

Sampling with multiple branches. A single patch is often insufficient. To draw K patches from an image, a popular solution is to sample K times from the same distribution without replacement [59, 65]. However, this may produce mutually similar patches, as shown in Figure 3a. To capture diverse patterns in the feature map, we propose a patch sampler with K branches. Each branch is designed to separately learn a different distribution and sample a patch from it. Let $(p_{i,r}, p_{i,r}^o)$ denote the patch features sampled by the r-th branch, then our patch sampler samples K patches for an image:

$$\left\{(p_{i,1}, p_{i,1}^o), \ldots, (p_{i,K}, p_{i,K}^o)\right\}. \tag{7}$$

This multi-branch design allows more diversity among sampled patches, as shown in Figure 3.


(a) Patch sampling with a single branch

(b) Patch sampling with K branches

Figure 3: Comparison of sampled patches on the CUHK03 dataset [32] with either single or multiple branches in the proposed model. Sampling locations are mapped back onto the image for better visualization.

Confidence loss. In order to guide the differentiable patch sampler to minimize patch-level distribution gap, the loss function has to penalize patches that are far from previous distributions. This may be seen as out-of-distribution detection at patch-level, and thus can be solved similarly. Inspired by a group of out-of-distribution detection methods that interpret inputs with low prediction confidence as out-of-distribution examples [19, 29, 35], we estimate each patch's closeness to previous distributions with its confidence on the old model. Given a patch with features $(p_{i,r}, p_{i,r}^o)$, its confidence on the old model can be measured by the negative entropy of the model prediction $\sigma(\phi^o(p_{i,r}^o))$ [44]. Based on this, we define the confidence loss in the form of entropy:

$$\mathcal{L}_{conf} = \frac{1}{BK} \sum_{i=1}^{B} \sum_{i=1}^{K} \mathrm{H}\left(\sigma(\phi^o(p_{i,r}^o))\right). \tag{8}$$

It is easy to see that minimizing $\mathcal{L}_{conf}$ is equivalent to maximizing the confidence of sampled patches on the old model, thereby training the sampler to select patches that are closer to previous distributions.

Diversity loss. To avoid the scores in K branches from converging to the same or similar values, resulting in poor patch diversity, we introduce a diversity loss to penalize mutually similar patch pairs, such as those with high cosine similarity:

$$\mathcal{L}_{div} = \frac{1}{BK^2} \sum_{i=1}^{B} \sum_{r,s=1}^{K} \frac{\langle p_{i,r}, p_{i,s} \rangle}{\|p_{i,r}\| \|p_{i,s}\|}, \tag{9}$$

where $\langle \cdot, \cdot \rangle$ is the inner product and $\|\cdot\|$ is the ℓ_2 norm. However, the diversity loss may distract the patch sampler away from its original objective. To cope with it, we introduce a weight γ_1 to control the degree of patch diversity, and define the total loss function for the patch sampler as

$$\mathcal{L}_{sel} = \mathcal{L}_{conf} + \gamma_1 \mathcal{L}_{div}. \tag{10}$$

4.3 Patch Logit Distillation

Patch logit distillation intends to utilize selected patch features for knowledge distillation. However, directly passing patch-level information would disturb image-level feature learning due to the distribution discrepancy between images and patches. For example, patch features can interfere with the batch statics in the classifier ϕ and then further affect the training of the whole model.

Instead of using a shared classifier ϕ, we employ a separate patch classifier ϕ_p to predict logits from patch features and distill knowledge from the old model. The distillation loss for all patches within a mini-batch is calculated as

$$\mathcal{L}_{PLD} = \frac{1}{BK} \sum_{i=1}^{B} \sum_{r=1}^{K} \mathrm{KL}\left(\sigma(\phi^o(p^o_{i,r})/\tau) \,\|\, \sigma(\phi_p(p_{i,r})/\tau) \right). \tag{11}$$

Since the patches are selected to be closer to the previous distributions, patch logit distillation is less affected by the distribution shift and thus more effective. When performing together with image-based distillation, it also helps retain more detailed knowledge about local cues.

4.4 Patch Relation Distillation

Relation distillation is also utilized to preserve high-order knowledge beyond class scores. The intuition is that knowledge can be complementarily represented by feature relations beyond individual features [42], which is in line with the task goal of person ReID (namely matching images for the same identity). While its power in lifelong learning has only recently been explored [8, 58], we take a step further by distilling on patch relations.

Among BK sampled patches within a mini-batch, there are a small proportion of intra-instance relations that correspond to local correlations within the image, and numerous inter-instance relations containing sparse yet valuable global identity information. Since intra-instance relations are significantly outnumbered, we propose to handle them separately. Suppose the patch relation is represented by pairwise feature distance $d(\cdot,\cdot)$, we consider the following two sets of distances, *i.e.*, all intra-instance distances and a fraction of inter-instance distances from the same sampler branch:

$$\begin{aligned} S_{intra} &= \bigcup_{i=1}^{B} \left\{ (d(p_{i,r}, p_{i,s}), d(p^o_{i,r}, p^o_{i,s})) \mid r, s \in [1\,..\,K], r \neq s \right\}, \\ S_{inter} &= \bigcup_{r=1}^{K} \left\{ (d(p_{i,r}, p_{j,r}), d(p^o_{i,r}, p^o_{j,r})) \mid i, j \in [1\,..\,B], i \neq j \right\}. \end{aligned} \tag{12}$$

It is desirable for the current model to be consistent with the old one in these distances, so we adopt a Huber loss l_δ to penalize the difference between each distance generated by the two models:

$$\mathcal{L}_{PRD} = \frac{\sum l_\delta(d_{intra} - d^o_{intra})}{|S_{intra}|} + \frac{\sum l_\delta(d_{inter} - d^o_{inter})}{|S_{inter}|}, \tag{13}$$

where $(d_{intra}, d^o_{intra}) \in S_{intra}$ and $(d_{inter}, d^o_{inter}) \in S_{inter}$. Since the intra-instance and inter-instance terms contribute equally to the distillation loss, both intra-instance and inter-instance relational knowledge are preserved during training.

The final loss function for our framework is

$$\mathcal{L} = \mathcal{L}_{base} + \mathcal{L}_{sel} + \gamma_2 \mathcal{L}_{PLD} + \gamma_3 \mathcal{L}_{PRD}, \tag{14}$$

where γ_2 and γ_3 are hyperparameters to balance the contributions of patch logit distillation and relation distillation, respectively.

Remarks. The patch sampler, the patch classifier and the patch relation distillation module are employed only to regularize the training of the ReID model and do not participate in the model inference during testing. Therefore, the extra computational overhead brought by our framework stays in the training stage and does not affect testing.

Table 1: Dataset statistics of the LReID benchmark [46]. Since the sampling procedure results in the numbers of train IDs being all 500, the original numbers of IDs are listed for comparison. '-' denotes that the dataset is not used for training.

Type	Dataset	Scale	#train IDs	#test IDs
Seen	Market-1501 [73]	large	500 (751)	750
	CUHK-SYSU [64]	mid	500 (942)	2900
	DukeMTMC-reID [48]	large	500 (702)	1110
	MSMT17_V2 [61]	large	500 (1041)	3060
	CUHK03 [32]	mid	500 (700)	700
Unseen	VIPeR [16]	small	-	316
	PRID [22]	small	-	649
	GRID [39]	small	-	126
	i-LIDS [74]	small	-	60
	CUHK01 [31]	small	-	486
	CUHK02 [30]	mid	-	239
	SenseReID [70]	mid	-	1718

5 EXPERIMENTS

5.1 Datasets and Evaluation Metrics

Datasets. We conduct extensive experiments on the LReID benchmark [46]. It consists of twelve ReID datasets, including five seen datasets (Market-1501 [73], CUHK-SYSU [64], DukeMTMC-reID [48], MSMT17_V2 [61] and CUHK03 [32]) for non-forgetting evaluation and seven unseen datasets (VIPeR [16], PRID [22], GRID [39], i-LIDS [74], CUHK01 [31], CUHK02 [30] and SenseReID [70]) for generalization evaluation. Note that CUHK-SYSU is a person search dataset, so we follow the procedure in [46] by first cropping the images using the ground-truth person bounding box annotation and then selecting a subset of identities with more than 4 bounding boxes. To mitigate the problem of unbalanced class number among datasets, we randomly sample 500 identities from each seen dataset for training. During evaluation, all test datasets are merged into one benchmark. To sum up, the processed ReID datasets contain 40459 training images of 2500 identities and 9854 testing images of 3594 identities in total. More detailed statistics for these datasets are provided in Table 1.

Table 2: Comparison with the state-of-the-art methods on the LReID benchmark. ‘*’ represents the base model in Section 4.1. ‘†’ represents the results reported in [46]. All methods adopt ResNet-50 as the feature-extracting backbone.

(a) Training order-1: Market-1501 → CUHK-SYSU → DukeMTMC-reID → MSMT17_V2 → CUHK03.

Method	Market-1501		CUHK-SYSU		DukeMTMC-reID		MSMT17_V2		CUHK03		Seen-Avg		Unseen-Avg	
	mAP	R1	mAP	R1	mAP	R1	mAP	R1	mAP	R1	mAP	R1	mAP	R1
Finetune	32.71	58.34	57.99	60.62	25.16	43.81	4.48	13.12	41.33	43.43	32.34	43.86	38.38	34.43
SPD [60]	35.63	61.16	61.70	63.97	27.45	47.13	5.18	15.45	**42.17**	**44.29**	34.43	46.40	40.38	36.57
LwF [34]*	56.27	77.11	72.86	75.14	29.59	46.45	5.99	16.55	36.10	37.50	40.16	50.55	47.16	42.57
CRL [69]	58.04	78.18	72.51	75.10	28.29	45.15	6.00	15.81	37.39	39.79	40.45	50.81	47.76	43.47
AKA [46]†	51.2	72.0	47.5	45.1	18.7	33.1	**16.4**	**37.6**	27.7	27.6	32.3	43.1	44.3	40.4
AKA [46]	58.05	77.43	72.52	74.83	28.65	45.15	6.13	16.22	38.66	40.43	40.80	50.81	47.60	42.63
Ours	**68.47**	**85.72**	**75.59**	**78.59**	**33.77**	**50.40**	6.49	16.96	34.11	36.79	**43.69**	**53.69**	**49.09**	**45.42**
JointTrain	68.12	85.24	81.35	83.83	60.36	75.72	24.57	48.87	42.74	43.57	55.43	67.45	49.82	46.29

(b) Training order-2: DukeMTMC-reID → MSMT17_V2 → Market-1501 → CUHK-SYSU → CUHK03.

Method	DukeMTMC-reID		MSMT17_V2		Market-1501		CUHK-SYSU		CUHK03		Seen-Avg		Unseen-Avg	
	mAP	R1	mAP	R1	mAP	R1	mAP	R1	mAP	R1	mAP	R1	mAP	R1
Finetune	26.12	45.74	3.31	10.30	29.12	54.10	57.20	60.03	40.34	40.93	31.22	42.22	36.10	32.04
SPD [60]	28.54	48.47	3.67	11.51	32.26	57.36	62.06	64.97	**43.00**	**45.21**	33.90	45.51	39.82	36.31
LwF [34]*	42.71	61.67	5.06	14.33	34.42	58.58	69.93	73.00	34.08	34.14	37.24	48.35	43.95	40.10
CRL [69]	43.47	63.11	4.81	13.69	35.03	59.77	70.01	72.79	34.49	36.79	37.56	49.23	45.28	41.43
AKA [46]†	32.5	49.7	-	-	-	-	-	-	-	-	-	-	40.8	37.2
AKA [46]	42.22	60.14	5.40	15.14	37.20	59.77	71.24	73.90	36.92	37.86	38.60	49.36	46.00	41.72
Ours	**58.27**	**74.10**	**6.39**	**17.39**	**43.18**	**67.40**	**74.52**	**76.90**	33.66	34.79	**43.20**	**54.11**	**48.60**	**44.12**
JointTrain	60.36	75.72	24.57	48.87	68.12	85.24	81.35	83.83	42.74	43.57	55.43	67.45	49.82	46.29

In order to simulate real-world lifelong learning scenarios with arbitrary domain order, we investigate two representative training orders used in [46]. Let training order-1 and order-2 represent Market-1501 → CUHK-SYSU → DukeMTMC-reID → MSMT17_V2 → CUHK03 and DukeMTMC-reID → MSMT17_V2 → Market-1501 → CUHK-SYSU → CUHK03, respectively.

Evaluation metrics. We use mean Average Precision (mAP) and Rank-1 accuracy (R1) to evaluate the performance on each ReID dataset. Moreover, the average performance on both seen datasets and unseen datasets are calculated as measures of non-forgetting performance and generalization ability, respectively.

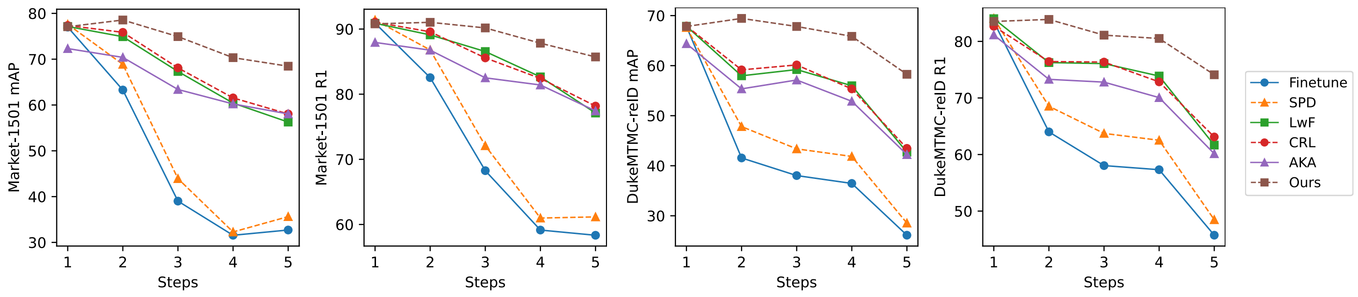
5.2 Implementation Details

We utilize a ResNet-50 [17] pretrained on ImageNet [50] as the feature extractor. Note that the last stride is set to 1 as suggested in [40]. Following [46], the model is trained for 50 epochs with 150 iterations per epoch using an Adam optimizer [27]. The learning rate is set to 3.5×10^{-4} initially and decays by ×0.1 at 25th and 35th epochs. The batch size is set to $B = 128$. In specific, each batch is composed of 32 identities and 4 images per identity. The input images are resized to 256×128 with data augmentations including random cropping, horizontal flipping and erasing. For the patch sampling step, patch features are sampled from the last feature map of size 16×8. The scorer for predicting sampling probabilities is a two-layer perceptron with 4096 hidden units.

We follow Zhao *et al.* [69] to set the loss weight $\gamma_0 = 1$ and the temperature $\tau = 2$. The number of patches per image and the remaining loss weights are empirically set as $K = 3$, $\gamma_1 = 0.5$, $\gamma_2 = 0.1$ and $\gamma_3 = 100$. We provide sensitivity analysis of newly introduced hyperparameters K, γ_1, γ_2 and γ_3 in Section 5.4. The whole architecture is implemented with PyTorch [43] and trained on single NVIDIA 2080 Ti GPU. During evaluation, the retrieval result is computed based on the Euclidean distance of image-level features, following the practice in [46].

5.3 Comparison with State-of-the-Art

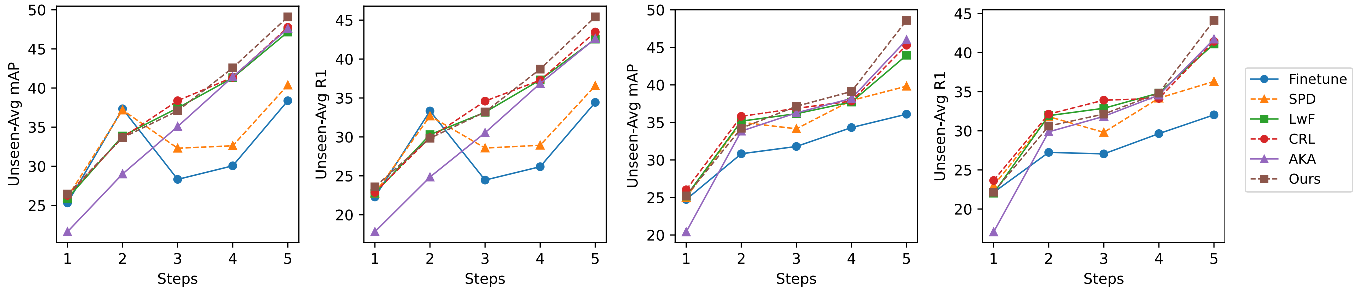
In this section, we compare our method to five lifelong learning methods that do not rely on exemplar memory: Finetune, SPD [60], LwF [34], CRL [69] and AKA [46]. Finetune denotes fine-tuning model on new datasets without knowledge distillation. SPD is an advanced feature distillation method. LwF, CRL and AKA are competitors based on logit distillation. For a fair comparison, these methods are reproduced with the same backbone and the same ReID loss consisting of cross-entropy loss and triplet loss, except AKA which uses its proposed plasticity loss instead of triplet loss. We summarize the final result of each method on the LReID benchmark in Table 2. We also report the upper-bound for each setting



(a) Training order-1

(b) Training order-2

Figure 4: Evolution of non-forgetting performance on the first seen dataset during training process.



(a) Training order-1

(b) Training order-2

Figure 5: Evolution of generalization ability on unseen datasets during training process.

estimated by JointTrain, where all data from all steps are assembled in advance for joint training once. The number of epochs for JointTrain is set to 250 to match the total training time.

Non-forgetting performance on seen datasets. Table 2 shows that on seen datasets, we improve the average mAP of state-of-the-art methods by 2.89% and 4.60% under training order-1 and order-2, respectively. In particular, on the Market-1501 dataset, the performance gap to the upper-bound in terms of mAP is almost eliminated (v.s. previous best with a gap of 10.07%) under training order-1 and narrowed from 16.89% to 2.09% under training order-2. Note that models with strong regularization to mitigate catastrophic forgetting usually have limited adaptation ability, *i.e.*, weak performance on subsequent datasets, while our method strikes a balance between non-forgetting and adaptation, thereby achieving competitive results on most seen datasets. Figure 4 illustrates the mAP and R1 curves on the first seen dataset after each training step. It can be seen that the forgetting of our method is moderate, and the performance is significantly better than that of alternative methods. This demonstrates that our method is effective for alleviating catastrophic forgetting in lifelong person ReID.

Generalization ability on unseen datasets. As shown in Table 2, our method outperforms the state-of-the-art method AKA on unseen datasets under both training orders. Specifically, we improve the average mAP by 1.49% under training order-1 and 2.60% under training order-2, further approaching the upper-bound. Figure 5 depicts the trend of average mAP and R1 on all unseen datasets during training process. We can observe that our method achieves the overall best performance. While some methods without distillation loss or with weaker distillation loss such as FineTune and CRL outperform us in the early steps, they fail to continuously improve their generalization ability. In contrast, our method shows consistent improvement over time.

Table 3: Ablation study of the patch sampler.

Method	Seen-Avg		Unseen-Avg	
	mAP	R1	mAP	R1
Base model	40.16	50.55	47.16	42.57
Ours w/o $\mathcal{L}_{conf}$	41.94	52.22	47.62	43.57
Ours w/o $\mathcal{L}_{div}$	42.90	52.60	47.71	43.60
Ours w/o multi-branch	43.13	52.79	48.32	43.67
Ours	**43.69**	**53.69**	**49.09**	**45.42**

5.4 Ablation Studies

In this section, we conduct experiments under training order-1 to examine the effectiveness of each module and the influence of hyperparameters.

Effectiveness of differentiable patch sampler. We introduce a differentiable patch sampler with three key components: a confidence loss, a diversity loss and a multi-branch design. Here, we analyze the impact of each component on the final performance in Table 3.

Table 4: Ablation study of patch-based distillation losses. 'ϕ' denotes using a shared classifier ϕ for patch logit distillation. 'FC' represents distilling knowledge of all patch relations in a unified way.

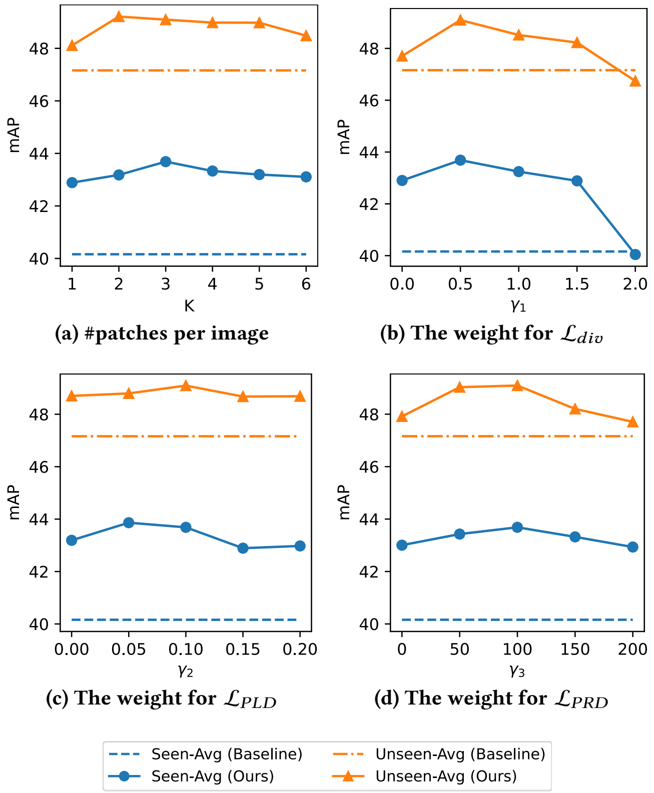
Losses		Seen-Avg		Unseen-Avg	
$\mathcal{L}_{PLD}$	$\mathcal{L}_{PRD}$	mAP	R1	mAP	R1
		40.16	50.55	47.16	42.57
ϕ		37.02	48.17	44.44	39.94
✓		43.00	52.98	47.91	44.02
✓	FC	43.54	**53.98**	48.42	43.47
✓	✓	**43.69**	53.69	**49.09**	**45.42**

It can be seen that the sampler trained without the confidence loss yields the second-lowest results only after the base model, suggesting that the guidance from the old model is crucial to our sampler. The diversity loss and the multi-branch design contribute to the final performance by helping to focus on diverse cues. For example, the multi-branch design allows each patch to be sampled from a different distribution, giving it an edge in terms of patch diversity compared to the sampling without replacement strategy [65] as previously shown in Figure 3. As a result, the full model is able to capture various local information and thus obtains the best performance.

Effectiveness of patch logit distillation. Table 4 explores the contribution of each patch-based distillation loss by adding the proposed losses. We can observe that using a shared classifier for patch logit distillation results in undesired performance degradation, while employing a separate patch classifier in our formulation yields better results. This validates our design in handling image-patch discrepancy. Compared to the base model, patch logit distillation brings a significant improvement by 2.84% and 0.75% in average mAP on seen and unseen datasets, respectively, verifying its high effectiveness.

Effectiveness of patch relation distillation. We compare the proposed patch relation distillation with a more straightforward method that distills on entire relations at once. As shown in Table 4, there is already a noticeable improvement by simply taking all patch relations into account, which is in line with the intuition that knowledge is complementarily represented by feature relationships. Treating intra-instance and inter-instance relations separately enables our model to achieve higher overall performance. Particularly, the average mAP on unseen datasets is lifted by 1.18% on top of patch logit distillation, indicating that patch relation distillation plays a key role in improving the generalization ability.

Hyperparameter analysis. In Figure 6, we vary newly introduced hyperparameters, namely, the number of patches per image K and the loss weights γ_1, γ_2 and γ_3, to study the sensitivity of our method to patch complexity, patch diversity and the two patch-based distillation modules. As shown in Figure 6a, the model produces relatively lower results when $K = 1$, since it can not access intra-instance patch pairs for relation distillation. On the other hand, excessively increasing patch complexity with a large K also leads to slightly weaker performance. Figure 6b demonstrates that γ_1 in the range of $[0.5, 1.5]$ benefits the final performance by ensuring a certain degree of patch diversity, while a larger γ_1 results in performance degradation as it distracts the patch sampler away from its original objective. Comparing Figure 6c and Figure 6d, we can observe that patch logit distillation has a larger impact on the non-forgetting performance, while patch relation distillation shows a greater influence on the generalization ability. Overall, the performance of our method is stable for these hyperparameters over a reasonably wide interval.



(a) #patches per image

(b) The weight for $\mathcal{L}_{div}$

(c) The weight for $\mathcal{L}_{PLD}$

(d) The weight for $\mathcal{L}_{PRD}$

Figure 6: Sensitivity to hyperparameters.

6 CONCLUSION

We address catastrophic forgetting in lifelong person ReID with the proposed patch-based knowledge distillation framework. It consists of a differentiable patch sampler, patch logit distillation and patch relation distillation. Specifically, the patch sampler is trained to select patches that are less affected by distribution shift with guidance from the old model. Patch logit distillation regularizes the current model to mimic the old one's prediction on the selected patches, while patch relation distillation preserves relational knowledge by imposing consistency constraints on intra-instance and inter-instance patch distances. Extensive experiments on the LReID benchmark demonstrate that our method outperforms state-of-the-art baselines in both non-forgetting and generalization evaluations.

Acknowledgement: This work is supported by National Key R&D Program of China (2020AAA0104401, 2018AAA0100702) and Beijing Natural Science Foundation (Z190001).

REFERENCES

[1] Rahaf Aljundi, Francesca Babiloni, Mohamed Elhoseiny, Marcus Rohrbach, and Tinne Tuytelaars. 2018. Memory aware synapses: Learning what (not) to forget. In *ECCV*. 139–154.

[2] Rahaf Aljundi, Punarjay Chakravarty, and Tinne Tuytelaars. 2017. Expert gate: Lifelong learning with a network of experts. In *CVPR*. 3366–3375.

[3] Jimmy Ba and Rich Caruana. 2014. Do deep nets really need to be deep?. In *NeurIPS*. 2654–2662.

[4] Yoshua Bengio, Nicholas Léonard, and Aaron Courville. 2013. Estimating or propagating gradients through stochastic neurons for conditional computation. *arXiv preprint arXiv:1308.3432* (2013).

[5] Quentin Berthet, Mathieu Blondel, Olivier Teboul, Marco Cuturi, Jean-Philippe Vert, and Francis Bach. 2020. Learning with differentiable pertubed optimizers. In *NeurIPS*. 9508–9519.

[6] Francisco M Castro, Manuel J Marín-Jiménez, Nicolás Guil, Cordelia Schmid, and Karteek Alahari. 2018. End-to-end incremental learning. In *ECCV*. 233–248.

[7] Fabio Cermelli, Massimiliano Mancini, Samuel Rota Bulo, Elisa Ricci, and Barbara Caputo. 2020. Modeling the background for incremental learning in semantic segmentation. In *CVPR*. 9233–9242.

[8] Hyuntak Cha, Jaeho Lee, and Jinwoo Shin. 2021. Co2l: Contrastive continual learning. In *ICCV*. 9516–9525.

[9] Hao Chen, Benoit Lagadec, and Francois Bremond. 2021. Ice: Inter-instance contrastive encoding for unsupervised person re-identification. In *ICCV*. 14960–14969.

[10] Jean-Baptiste Cordonnier, Aravindh Mahendran, Alexey Dosovitskiy, Dirk Weissenborn, Jakob Uszkoreit, and Thomas Unterthiner. 2021. Differentiable patch selection for image recognition. In *CVPR*. 2351–2360.

[11] Corinna Cortes, Xavier Gonzalvo, Vitaly Kuznetsov, Mehryar Mohri, and Scott Yang. 2017. Adanet: Adaptive structural learning of artificial neural networks. In *ICML*. 874–883.

[12] Yongxing Dai, Jun Liu, Yifan Sun, Zekun Tong, Chi Zhang, and Ling-Yu Duan. 2021. Idm: An intermediate domain module for domain adaptive person re-id. In *ICCV*. 11864–11874.

[13] Matthias Delange, Rahaf Aljundi, Marc Masana, Sarah Parisot, Xu Jia, Ales Leonardis, Greg Slabaugh, and Tinne Tuytelaars. 2021. A continual learning survey: Defying forgetting in classification tasks. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2021), 1–1.

[14] Prithviraj Dhar, Rajat Vikram Singh, Kuan-Chuan Peng, Ziyan Wu, and Rama Chellappa. 2019. Learning without memorizing. In *CVPR*. 5138–5146.

[15] Yixiao Ge, Dapeng Chen, and Hongsheng Li. 2020. Mutual mean-teaching: Pseudo label refinery for unsupervised domain adaptation on person re-identification. In *ICLR*.

[16] Douglas Gray and Hai Tao. 2008. Viewpoint invariant pedestrian recognition with an ensemble of localized features. In *ECCV*. 262–275.

[17] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *CVPR*. 770–778.

[18] Shuting He, Hao Luo, Pichao Wang, Fan Wang, Hao Li, and Wei Jiang. 2021. Transreid: Transformer-based object re-identification. In *ICCV*. 15013–15022.

[19] Dan Hendrycks and Kevin Gimpel. 2017. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *ICLR*.

[20] Alexander Hermans, Lucas Beyer, and Bastian Leibe. 2017. In defense of the triplet loss for person re-identification. *arXiv preprint arXiv:1703.07737* (2017).

[21] Geoffrey Hinton, Oriol Vinyals, Jeff Dean, et al. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531* (2015).

[22] Martin Hirzer, Csaba Beleznai, Peter M Roth, and Horst Bischof. 2011. Person re-identification by descriptive and discriminative classification. In *Scandinavian conference on Image analysis*. 91–102.

[23] Max Jaderberg, Karen Simonyan, Andrew Zisserman, et al. 2015. Spatial transformer networks. In *NeurIPS*. 2017–2025.

[24] Eric Jang, Shixiang Gu, and Ben Poole. 2017. Categorical reparameterization with gumbel-softmax. In *ICLR*.

[25] Angelos Katharopoulos and François Fleuret. 2019. Processing megapixel images with deep attention-sampling models. In *ICML*. 3282–3291.

[26] Youmin Kim, Jinbae Park, YounHo Jang, Muhammad Ali, Tae-Hyun Oh, and Sung-Ho Bae. 2021. Distilling Global and Local Logits With Densely Connected Relations. In *ICCV*. 6290–6300.

[27] Diederik P Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *ICLR*.

[28] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. 2017. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences* 114, 13 (2017), 3521–3526.

[29] Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. 2018. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. In *NeurIPS*. 7167–7177.

[30] Wei Li and Xiaogang Wang. 2013. Locally aligned feature transforms across views. In *CVPR*. 3594–3601.

[31] Wei Li, Rui Zhao, and Xiaogang Wang. 2012. Human reidentification with transferred metric learning. In *ACCV*. 31–44.

[32] Wei Li, Rui Zhao, Tong Xiao, and Xiaogang Wang. 2014. Deepreid: Deep filter pairing neural network for person re-identification. In *CVPR*. 152–159.

[33] Xiaojie Li, Jianlong Wu, Hongyu Fang, Yue Liao, Fei Wang, and Chen Qian. 2020. Local correlation consistency for knowledge distillation. In *ECCV*. 18–33.

[34] Zhizhong Li and Derek Hoiem. 2017. Learning without forgetting. *IEEE transactions on pattern analysis and machine intelligence* 40, 12 (2017), 2935–2947.

[35] Shiyu Liang, Yixuan Li, and Rayadurgam Srikant. 2018. Enhancing the reliability of out-of-distribution image detection in neural networks. In *ICLR*.

[36] Yutian Lin, Xuanyi Dong, Liang Zheng, Yan Yan, and Yi Yang. 2019. A bottom-up clustering approach to unsupervised person re-identification. In *AAAI*. 8738–8745.

[37] Yufan Liu, Jiajiong Cao, Bing Li, Chunfeng Yuan, Weiming Hu, Yangxi Li, and Yunqiang Duan. 2019. Knowledge distillation via instance relationship graph. In *CVPR*. 7096–7104.

[38] David Lopez-Paz and Marc'Aurelio Ranzato. 2017. Gradient episodic memory for continual learning. In *NeurIPS*. 6470–6479.

[39] Chen Change Loy, Tao Xiang, and Shaogang Gong. 2010. Time-delayed correlation analysis for multi-camera activity understanding. *International Journal of Computer Vision* 90, 1 (2010), 106–129.

[40] Hao Luo, Youzhi Gu, Xingyu Liao, Shenqi Lai, and Wei Jiang. 2019. Bag of tricks and a strong baseline for deep person re-identification. In *CVPR Workshops*.

[41] Arun Mallya and Svetlana Lazebnik. 2018. Packnet: Adding multiple tasks to a single network by iterative pruning. In *CVPR*. 7765–7773.

[42] Wonpyo Park, Dongju Kim, Yan Lu, and Minsu Cho. 2019. Relational knowledge distillation. In *CVPR*. 3967–3976.

[43] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. Pytorch: An imperative style, high-performance deep learning library. In *NeurIPS*. 8024–8035.

[44] Tim Pearce, Alexandra Brintrup, and Jun Zhu. 2021. Understanding softmax confidence and uncertainty. *arXiv preprint arXiv:2106.04972* (2021).

[45] Baoyun Peng, Xiao Jin, Jiaheng Liu, Dongsheng Li, Yichao Wu, Yu Liu, Shunfeng Zhou, and Zhaoning Zhang. 2019. Correlation congruence for knowledge distillation. In *ICCV*. 5007–5016.

[46] Nan Pu, Wei Chen, Yu Liu, Erwin M Bakker, and Michael S Lew. 2021. Lifelong person re-identification via adaptive knowledge accumulation. In *CVPR*. 7901–7910.

[47] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. 2017. icarl: Incremental classifier and representation learning. In *CVPR*. 2001–2010.

[48] Ergys Ristani, Francesco Solera, Roger Zou, Rita Cucchiara, and Carlo Tomasi. 2016. Performance measures and a data set for multi-target, multi-camera tracking. In *ECCV Workshops*.

[49] Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. 2015. Fitnets: Hints for thin deep nets. In *ICLR*.

[50] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. 2015. Imagenet large scale visual recognition challenge. *International journal of computer vision* 115, 3 (2015), 211–252.

[51] Andrei A Rusu, Neil C Rabinowitz, Guillaume Desjardins, Hubert Soyer, James Kirkpatrick, Koray Kavukcuoglu, Razvan Pascanu, and Raia Hadsell. 2016. Progressive neural networks. *arXiv preprint arXiv:1606.04671* (2016).

[52] Yang Shen, Weiyao Lin, Junchi Yan, Mingliang Xu, Jianxin Wu, and Jingdong Wang. 2015. Person re-identification with correspondence structure learning. In *ICCV*. 3200–3208.

[53] Hanul Shin, Jung Kwon Lee, Jaehong Kim, and Jiwon Kim. 2017. Continual learning with deep generative replay. In *NeurIPS*. 2994–3003.

[54] Konstantin Shmelkov, Cordelia Schmid, and Karteek Alahari. 2017. Incremental learning of object detectors without catastrophic forgetting. In *ICCV*. 3400–3409.

[55] Chi Su, Jianing Li, Shiliang Zhang, Junliang Xing, Wen Gao, and Qi Tian. 2017. Pose-driven deep convolutional model for person re-identification. In *ICCV*. 3960–3969.

[56] Nehemia Sugianto, Dian Tjondronegoro, Golam Sorwar, Prithwi Chakraborty, and Elizabeth Irenne Yuwono. 2019. Continuous learning without forgetting for person re-identification. In *2019 16th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*. 1–8.

[57] Yifan Sun, Liang Zheng, Yi Yang, Qi Tian, and Shengjin Wang. 2018. Beyond part models: Person retrieval with refined part pooling (and a strong convolutional baseline). In *ECCV*. 480–496.

[58] Xiaoyu Tao, Xinyuan Chang, Xiaopeng Hong, Xing Wei, and Yihong Gong. 2020. Topology-preserving class-incremental learning. In *ECCV*. 254–270.

[59] Hung-Yu Tseng, Hsin-Ying Lee, Lu Jiang, Ming-Hsuan Yang, and Weilong Yang. 2020. Retrievegan: Image synthesis via differentiable patch retrieval. In *ECCV*. 242–257.

[60] Frederick Tung and Greg Mori. 2019. Similarity-preserving knowledge distillation. In *ICCV*. 1365–1374.

[61] Longhui Wei, Shiliang Zhang, Wen Gao, and Qi Tian. 2018. Person transfer gan to bridge domain gap for person re-identification. In *CVPR*. 79–88.

[62] Chenshen Wu, Luis Herranz, Xialei Liu, Joost van de Weijer, Bogdan Raducanu, et al. 2018. Memory replay gans: Learning to generate new categories without forgetting. In *NeurIPS*. 5966–5976.

[63] Guile Wu and Shaogang Gong. 2021. Generalising without forgetting for lifelong person re-identification. In *AAAI*. 2889–2897.

[64] Tong Xiao, Shuang Li, Bochao Wang, Liang Lin, and Xiaogang Wang. 2016. End-to-end deep learning for person search. *arXiv preprint arXiv:1604.01850* (2016).

[65] Sang Michael Xie and Stefano Ermon. 2019. Reparameterizable subset sampling via continuous relaxations. In *IJCAI*.

[66] Qize Yang, Hong-Xing Yu, Ancong Wu, and Wei-Shi Zheng. 2019. Patch-based discriminative feature learning for unsupervised person re-identification. In *CVPR*. 3633–3642.

[67] Sergey Zagoruyko and Nikos Komodakis. 2016. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. In *ICLR*.

[68] Friedemann Zenke, Ben Poole, and Surya Ganguli. 2017. Continual learning through synaptic intelligence. In *ICML*. 3987–3995.

[69] Bo Zhao, Shixiang Tang, Dapeng Chen, Hakan Bilen, and Rui Zhao. 2021. Continual representation learning for biometric identification. In *2021 IEEE Winter Conference on Applications of Computer Vision (WACV)*. 1198–1208.

[70] Haiyu Zhao, Maoqing Tian, Shuyang Sun, Jing Shao, Junjie Yan, Shuai Yi, Xiaogang Wang, and Xiaoou Tang. 2017. Spindle net: Person re-identification with human body region guided feature decomposition and fusion. In *CVPR*. 1077–1085.

[71] Liming Zhao, Xi Li, Yueting Zhuang, and Jingdong Wang. 2017. Deeply-learned part-aligned representations for person re-identification. In *ICCV*. 3219–3228.

[72] Liang Zheng, Yujia Huang, Huchuan Lu, and Yi Yang. 2019. Pose-invariant embedding for deep person re-identification. *IEEE Transactions on Image Processing* 28, 9 (2019), 4500–4509.

[73] Liang Zheng, Liyue Shen, Lu Tian, Shengjin Wang, Jingdong Wang, and Qi Tian. 2015. Scalable person re-identification: A benchmark. In *ICCV*. 1116–1124.

[74] Wei-Shi Zheng, Shaogang Gong, and Tao Xiang. 2009. Associating Groups of People.. In *BMVC*. 1–11.

[75] Zhedong Zheng, Xiaodong Yang, Zhiding Yu, Liang Zheng, Yi Yang, and Jan Kautz. 2019. Joint discriminative and generative learning for person re-identification. In *CVPR*. 2138–2147.

[76] Zhun Zhong, Liang Zheng, Zhiming Luo, Shaozi Li, and Yi Yang. 2019. Invariance matters: Exemplar memory for domain adaptive person re-identification. In *CVPR*. 598–607.

Appendix

A NETWORK ARCHITECTURE

We summarize the network architectures of the proposed framework in Table 5.

Table 5: Network architectures of our proposed framework. The framework includes a feature extractor, a scorer and a classifier.

(a) The feature extractor. Its last stride (in conv5_1) is set to 1.

Layer name	Input size	Output size	Feature extractor
conv1	$3 \times 256 \times 128$	$64 \times 128 \times 64$	7×7, 64, stride 2
conv2_x	$64 \times 128 \times 64$	$64 \times 64 \times 32$	3×3 max pool, stride 2
	$64 \times 64 \times 32$	$256 \times 64 \times 32$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$
conv3_x	$256 \times 64 \times 32$	$512 \times 32 \times 16$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$
conv4_x	$512 \times 32 \times 16$	$1024 \times 16 \times 8$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 6$
conv5_x	$1024 \times 16 \times 8$	$2048 \times 16 \times 8$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$
avg_pool	$2048 \times 16 \times 8$	2048	global avg pool

(b) The scorer. The number of patches per image K is set to 3.

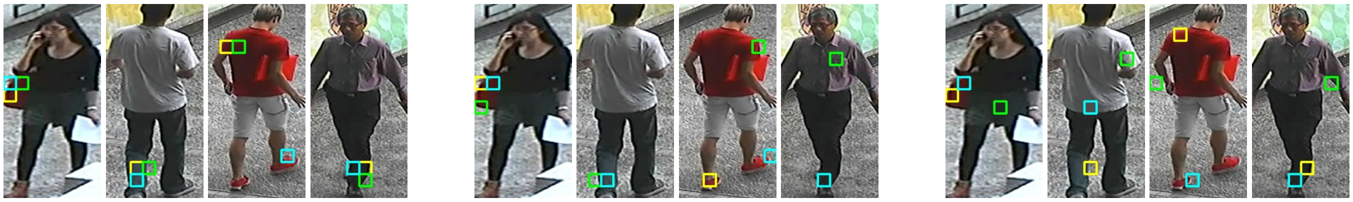
Layer name	Input size	Output size	Scorer
layer1	$2048 \times 16 \times 8$	$4096 \times 16 \times 8$	1×1, 4096, stride 1
layer2	$4096 \times 16 \times 8$	$3 \times 16 \times 8$	1×1, 3, stride 1

(c) The classifier. The class number is set to 2500 according to the LReID benchmark.

Layer name	Input size	Output size	Classifier
fc	2048	2500	2500-d fc

B PATCH VISUALIZATION

As shown in Figure 7, both the multi-branch design and the diversity loss $\mathcal{L}_{div}$ are essential to ensure patch diversity.

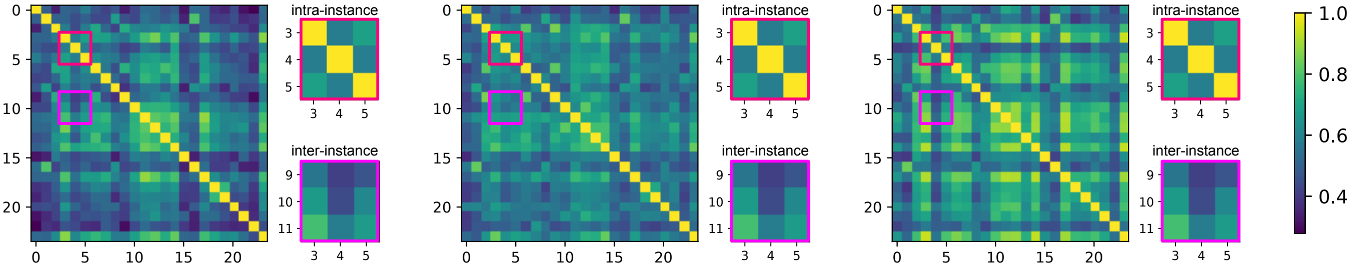


(a) Our method without multi-branch **(b) Our method without $\mathcal{L}_{div}$** **(c) Our full method**

Figure 7: Comparison of sampled patches on the CUHK03 dataset under training order-1. Sampling locations are mapped back onto the image for better visualization. Note that some patches in (b) are overlapped.

C PATCH RELATION VISUALIZATION

As demonstrated in Figure 8, patch relations are better preserved with the proposed distillation loss $\mathcal{L}_{PRD}$.



(a) The model before adaptation

(b) The model after adaptation using our method without $\mathcal{L}_{PRD}$

(c) The model after adaptation using our full method

Figure 8: Comparison of patch feature similarities on the first seen dataset under training-order 1. The patches are randomly sampled from a mini-batch of batch size $B = 8$. It can be seen that our full method preserves some key patch relations better than the method without $\mathcal{L}_{PRD}$.

D PATCH SIZE

The patch size in this paper is adopted as 16 × 16 in all experiments. We empirically find the size is already sufficient for achieving knowledge distillation in the person re-identificaiton task, as demonstrated by the experimental results in Table 2. Intuitively, the stacked convolutions in the model will gradually enlarge the receptive field of each neuron. As a result, even a single neuron (e.g., a neuron corresponding to the aforementioned 16 by 16 patch) at the topmost layer will admit non-local feature representation. Choosing the right patches is regarded to be more imporant than using a large patch size.