

Long Short-Term Knowledge Decomposition and Consolidation for Lifelong Person Re-Identification

Kunlun Xu , Zichen Liu , Xu Zou , *Member, IEEE*, Yuxin Peng , *Senior Member, IEEE*, and Jiahuan Zhou 

I. INTRODUCTION

Abstract—Lifelong person re-identification (LReID) aims to learn from streaming data sources step by step, which suffers from the catastrophic forgetting problem. In this paper, we investigate the exemplar-free LReID setting where no previous exemplar is available during the new step training. Existing exemplar-free LReID methods primarily adopt knowledge distillation to transfer knowledge from an old model to a new one without selection, inevitably introducing erroneous and detrimental information that hinders new knowledge learning. Furthermore, not all critical knowledge can be transferred due to the absence of old data, leading to the permanent loss of undistilled knowledge. To address these limitations, we propose a novel exemplar-free LReID method named Long Short-Term Knowledge Decomposition and Consolidation (LSTKC++). Specifically, an old knowledge rectification mechanism is developed to rectify the old model predictions based on new data annotations, ensuring correct knowledge transfer. Besides, a long-term knowledge consolidation strategy is designed, which first estimates the degree of old knowledge forgetting by leveraging the output difference between the old and new models. Then, a knowledge-guided parameter fusion strategy is developed to balance new and old knowledge, improving long-term knowledge retention. Upon these designs, considering LReID models tend to be biased on the latest seen domains, the fusion weights generated by this process often lead to sub-optimal knowledge balancing. To settle this, we further propose to decompose a single old model into two parts: a long-term old model containing multi-domain knowledge and a short-term model focusing on the latest short-term old knowledge. Then, the incoming new data are explored as an unbiased reference to adjust the old models' fusion weight to achieve backward optimization. Furthermore, an extended complementary knowledge rectification mechanism is developed to mine and retain the correct knowledge in the decomposed models. Extensive experimental results demonstrate that LSTKC++ significantly outperforms state-of-the-art methods by large margins.

Index Terms—Lifelong learning, person re-identification, knowledge distillation, knowledge rectification.

Received 8 September 2024; revised 18 March 2025; accepted 19 May 2025. Date of publication 22 May 2025; date of current version 6 August 2025. This work was supported in part by the National Natural Science Foundation of China under Grant 62376011, Grant 62525201, Grant 62132001, and Grant 62432001, in part by Beijing Natural Science Foundation under Grant L247006 and in part by the National Key R&D Program of China under Grant 2024YFA1410000. Recommended for acceptance by Y.-C. F. Wang. (*Corresponding author: Jiahuan Zhou.*)

Kunlun Xu, Zichen Liu, Yuxin Peng, and Jiahuan Zhou are with the Wangxuan Institute of Computer Technology, Peking University, Beijing 100080, China (e-mail: xkl@stu.pku.edu.cn; lzc20180720@stu.pku.edu.cn; pengyuxin@pku.edu.cn; jiahuanzhou@pku.edu.cn).

Xu Zou is with the School of Artificial Intelligence and Automation, Huazhong University of Science and Technology, Wuhan 430074, China (e-mail: zoux@hust.edu.cn).

The code is available at <https://github.com/zhoujiahuan1991/LSTKC++>.
Digital Object Identifier 10.1109/TPAMI.2025.3572468

PERSON re-identification (ReID) aims to retrieve the person of interest from a collection of images [1]. Numerous investigations [2], [3] have observed that ReID models trained on a specific dataset often perform poorly when confronted with new datasets. This limitation has sparked increasing interest in lifelong person re-identification (LReID), which focuses on continuously learning short-term knowledge from a stream of temporarily accessible datasets [4] to accomplish long-term knowledge accumulation. Similar to other lifelong learning tasks [5], the challenge of catastrophic forgetting emerges as a critical obstacle in LReID [6]. To handle this issue, several LReID approaches [7], [8], [9] aim to retain exemplars from the old datasets as the rehearsal of historical knowledge for the learning of new models. However, this solution inevitably impedes data privacy and suffers from heavy computational overheads. Therefore, in this paper, we concentrate on a more practical but challenging LReID setting where no exemplars can be preserved for new model learning, known as *exemplar-free LReID*.

In addition to exemplar preservation, a majority of existing LReID approaches [4], [9], [10] strive to alleviate catastrophic forgetting through knowledge distillation, primarily aimed at ensuring output consistency between the old and new models utilizing the given new data. However, as is shown in Fig. 1 (a), due to the domain gaps between different datasets, the old model can generate erroneous predictions on the new data because the long-term historical knowledge is biased and erroneous for the new domain. Indiscriminately distilling the knowledge from the old model to the new one will inevitably cause the accumulation of erroneous and detrimental knowledge, interfering with the new knowledge acquisition. Moreover, due to the absence of old data, it is hard to comprehensively distill all the informative knowledge from the old model, and the undistilled old knowledge is forgotten permanently. This oversight leads to severe forgetting of old knowledge in the long run.

To address the above issues, we propose a novel exemplar-free LReID approach termed Short-Term Knowledge Decomposition and Consolidation (LSTKC++). Our main idea is to design the LReID solution from two perspectives: During training with the new data, rectify the erroneous knowledge of the old model to avoid its interference in acquisition. Once the training is finished, the parameters of the new and old models are adaptively fused to improve long-term knowledge consolidation. Specifically, as illustrated in Fig. 1(b), by rectifying the output of the old model, the erroneous historical knowledge can be filtered out and

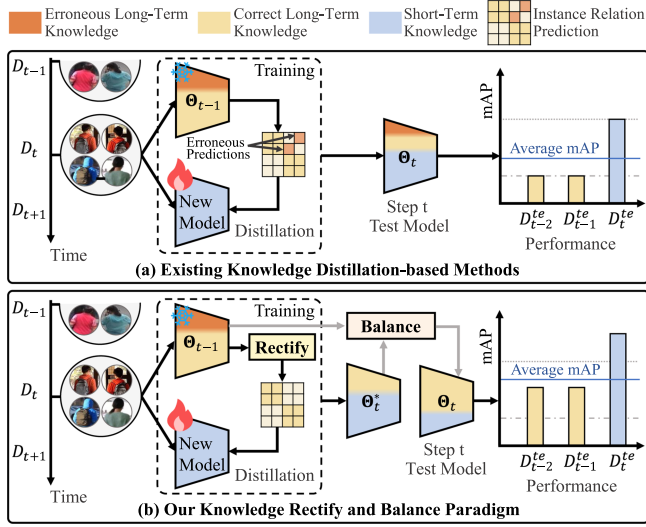


Fig. 1. (a) Existing LReID methods distill the long-term knowledge from the old model to the new model without selection, where the erroneous knowledge is also transferred to the new model, degrading the short-term new knowledge learning. Moreover, the undistilled long-term knowledge will be forgotten permanently. (b) Our proposed method rectifies the erroneous knowledge in the old model predictions, ensuring correct knowledge transfer to facilitate new knowledge learning. Furthermore, our approach actively balances the acquired long-term and short-term knowledge, enhancing average performance across diverse domains.

the learned new model will mainly contain correct knowledge. Besides, by balancing the old and new knowledge after the new model training, the historical and new knowledge can be better integrated to improve the average performance across various domains.

To achieve this, a Rectification-based Short-Term Knowledge Transfer (R-STKT) module is introduced to recognize and rectify the old model prediction based on the new data annotation. Besides, after the new model training, an Estimation-based Long-Term Knowledge Consolidation (E-LTKC) module is developed to estimate the degree of old knowledge forgetting based on the output difference between the old and new models. Then, a knowledge-guided model fusion strategy is designed to adaptively generate a fusion weight to balance the new and old knowledge. Furthermore, considering that LReID models tend to be biased on the latest seen domains and the old data are inaccessible, the old knowledge forgetting degree is hard to estimate, leading to suboptimal knowledge balancing. Therefore, we further aim to rectify the model fusion weight of the previous learning step to enhance long-term knowledge retention.

As shown in Fig. 2, we decompose a single old model into a long-term old model containing multi-domain knowledge and a short-term model primarily containing the short-term knowledge of the previous step. Based on the decomposition, an Old Knowledge Backward Rectification module is accordingly designed to utilize the new data as an unbiased reference to adjust the old models' fusion weight, achieving backward optimization of old knowledge consolidation. Furthermore, an extended Complementary Short-Term Transfer module is developed to mine the complementary information within the decomposed models, enhancing the retention of correct old knowledge. Moreover,

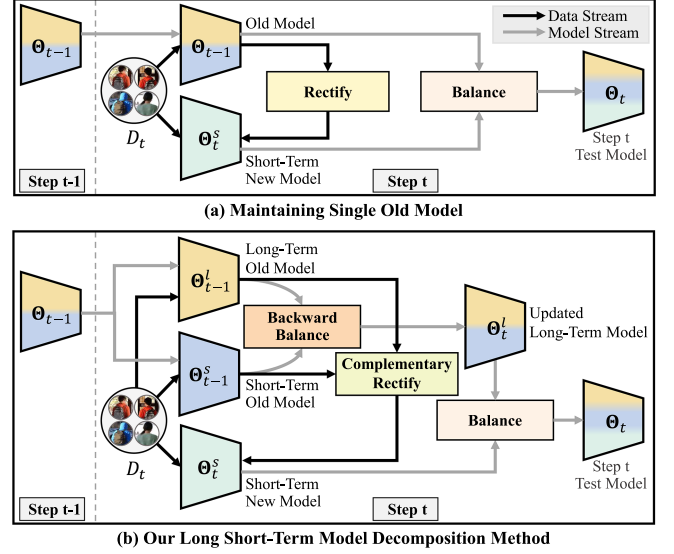


Fig. 2. (a) Maintaining a single old model is a popular choice that supports forward knowledge balance in our framework. (b) We propose to decompose a single old model into a long-term old model and a short-term old model. The backward balance is enabled to achieve better old knowledge consolidation. Besides, the complementary knowledge of the two decomposed old models can be mined to enrich the maintained correct knowledge.

we introduce a Relation Similarity-guided Long-Term Consolidation module that exploits new and old models to generate a robust inference model. Its effectiveness is demonstrated by theoretical analysis. Extensive experimental results show that our LSTKC++ significantly outperforms the state-of-the-art LReID methods on various benchmarks.

To sum up, the contributions of this paper are as follows: (1) A novel exemplar-free LReID method LSTKC++ is proposed, which introduces a new paradigm that conducts old knowledge rectification and adaptive knowledge balancing to mitigate catastrophic forgetting. (2) An old knowledge decomposition mechanism is introduced, based on which a backward rectification module is developed to optimize old knowledge consolidation and a complementary knowledge mining strategy is introduced to promote the utilization of correct old knowledge. (3) Thorough theoretical justifications and experimental analyses demonstrate the effectiveness of our proposed LReID model. (4) Extensive experiments demonstrate that our LSTKC++ requires less computational and storage overhead but outperforms the state-of-the-art LReID method by large margins.

This manuscript is an extension of our previous conference paper [11]. In this study, we have made extensive extensions including (1) Considering the limited knowledge balancing capacity when maintaining a single old model, we propose to decompose the single old model into a long-term old model and a short-term old model, where the long-term and short-term old knowledge are disentangled for more comprehensive utilization and consolidation. (2) To fully utilize the decomposed old models, an Old Knowledge Backward Rectification mechanism and a Complementary Short-Term Transfer module are developed to further enhance knowledge balancing and rectification capacity. Besides, a Relation Similarity-guided Long-Term Consolidation

module is introduced to obtain a robust inference model based on the available short-term and long-term models. (3) Comprehensive theoretical analyses are provided in Section IV, along with validation experiments to verify the effectiveness of our algorithm. (4) Beyond the two LReID orders evaluated in [11], we further evaluate our method on five additional challenging orders to demonstrate its robustness. (5) More ablation studies and detailed analyses, including learning trends, computational/storage resource overhead, and ablation studies on module effectiveness, are presented in Section V.

II. RELATED WORK

A. Person Re-Identification

Person Re-Identification (ReID) aims to justify whether the given images from distinct cameras, times, and locations contain the same person [12], [13], [14], [15], [16]. In the past twenty years, ReID has been extensively studied in a close setting, where the scenes and cameras of the test data are identical to the training ones [17], [18], [19], [20], [21]. While such a training setting yields strong performance on seen known scenes and cameras, it often exhibits significant performance degradation when applied to novel scenes and devices due to domain gaps, limiting the practical deployment of existing ReID models [22], [23], [24], [25]. To settle this, some works focused on the Unsupervised Domain Adaptation (UDA) problem [16], [26], [27], [28], [29], [30], [31], where a labeled source dataset and an unlabeled target dataset are given to update the model. However, UDA only considers adapting to a single target domain and neglects the practical scenario that new domains emerge continually in the surveillance systems. Therefore, the model should be equipped with the capacity to learn continually and perform well on both newly incoming and historically seen domains to handle the diverse test environments [4], [7]. Such a challenging and practical task is called Lifelong Person Re-Identification.

B. Lifelong Person Re-Identification

LReID has drawn increasing attention in recent years [4], [7], [8], [9], [11]. Similar to other lifelong learning tasks [32], [33], [34], the catastrophic forgetting problem is also the main challenge, which indicates that the model performance on historical datasets will degrade seriously when the model is trained on new ones [35], [36], [37], [38]. To overcome this problem, various methods [7], [8], [9], [39] have been proposed, which can be mainly categorized into two branches: data replay-based methods and knowledge distillation-based methods.

The data replay-based approaches aim to mitigate knowledge forgetting by storing and replaying exemplars from historical domains [7], [8], [9], [40]. GwFReID [7] reserved two exemplars for each seen identity and designed exemplar-based representation/distribution coherence constraints to force the model to maintain historical knowledge. PTKP [8] stored 500 images of 250 randomly selected identities for each seen domain. The exemplars are utilized to guide the model to mine inter-domain shared information. KRKC [9] was inspired by the biological process of human cognition and proposed an

exemplar-based knowledge refreshing scheme to accomplish knowledge consolidation. Although exemplar-based methods achieved remarkable performance on anti-forgetting [41], they tended to hinder data privacy and incur substantial computational and exemplar-storing overheads [42], [43]. Therefore, in this paper, we focus on the non-exemplar LReID setting to avoid privacy and memory/computing expansion issues.

The knowledge distillation-based (KD) methods [4], [10], [11], [42] tried to address the catastrophic forgetting problem by forcing the new model to generate consistent outputs with the old model. AKA [4] was one of the pioneering works that introduced Knowledge distillation into the LReID task. A logit distillation strategy was adopted to constrain the new model to generate the same classification distribution as the old model. PatchKD [10] designed a patch distillation module that can recognize the important regions of the image and distill the patch logits within such regions. MRN [42] emphasized the crucial role of Batch Normalization (BN) in data distribution shifting and proposed reconciliation normalization to constrain the learning of BN layers. Besides, a balanced knowledge distillation loss was proposed to handle the long-tailed distribution of person data, reinforcing the learned knowledge of tail identities. DKP [39] proposed the distribution-aware prototype to store the previous domain information and introduced a prototype-based knowledge distillation strategy to mitigate forgetting. CKP [44] focused on lifelong person re-identification in the presence of label noise. A weighted knowledge distillation loss is proposed to preserve clean knowledge from previous domains while actively forgetting noisy knowledge. Apart from the above non-exemplar LReID methods, some exemplar-based approaches, *e.g.*, PTKP [8] and KRKC [9], also adopt knowledge distillation as an auxiliary strategy to alleviate the catastrophic forgetting problem.

Although KD has been widely acknowledged as an effective anti-forgetting strategy, it suffers from two main drawbacks. 1) Due to the domain gaps, some knowledge acquired from the old model may be erroneous for the new domain. When erroneous knowledge is typically treated equally with correct knowledge during KD, the learning of the new model can be misled. 2) When employing KD strategies, only the knowledge that can be reflected in the new data is distilled. In contrast, the knowledge that cannot be manifested in the features of the new data is inevitably forgotten, causing a dramatic loss of long-term knowledge. In this paper, we address the above drawbacks by rectifying the erroneous historical knowledge during training and adaptively balancing long-term and short-term knowledge to achieve a better trade-off.

Different from data replay-based and knowledge distillation-based methods, the recently proposed PAEMA [45] introduced a prompt-based approach, incorporating a prompt difference-guided model reparameterization algorithm to balance knowledge acquisition and forgetting. However, as PAEMA is specifically designed for the vision transformer backbone, its adaptability to other architectures, particularly CNNs, is limited. In contrast, our proposed method is backbone-agnostic, making it applicable to a wide range of architectures.

C. Class Incremental Learning

Class incremental learning (CIL) [46], [47], [48], [49] is a popular lifelong learning task that aims to learn to recognize sequentially occurring new classes, which is similar to the LReID task that learns to recognize incoming new identities as the new domain shows up. Therefore, the CIL algorithms can be readily applied to the LReID task [4], [39], [50]. However, there exists a fundamental distinction between CIL and LReID. Specifically, during the testing phase, CIL models are evaluated using data from known classes [46], [51], whereas LReID models are evaluated on new identities that are not included in the training data [8]. Consequently, CIL approaches primarily focus on handling seen classes, whereas LReID methods emphasize the ability to distinguish novel identities. Therefore, CIL methods typically achieve suboptimal performance in the LReID task [4], [39]. In this paper, we investigate relation-based knowledge rectification and distillation, enhancing the retention of correct knowledge in the LReID task.

D. Knowledge Rectification

Knowledge rectification aims to filter and rectify erroneous knowledge in a given model or knowledge proxy [52], [53]. RKR [52] proposed to learn additional trainable parameters to rectify the weights of a pre-trained classification model. Liu et. al. [53] proposed to rectify the historical prototypes to diminish the cross-class bias instead of using fixed prototypes. The most similar work to this paper is DKR [54], which rectifies the image-text matching relation for continual cross-modal retrieval. Specifically, the one-to-one labeled pairs are utilized to recognize and rectify the erroneous matching knowledge. However, for LReID, each sample has a set of matching instances [1]. The one-to-set matching nature leads to a more complex inter-instance relation structure, requiring the algorithm to comprehensively consider the multiple inter-instance relations to prevent severe structure distortion.

III. PROPOSED METHOD

A. Problem Definition and Notations

In LReID, the training datasets from different domains are provided step by step. When the current data are obtained, the data of previous and later steps are unavailable [4], [10]. More specifically, the training data consists of a stream of T datasets $\mathcal{D} = \{D_t\}_{t=1}^T$ where each $D_t = \{(x_i, y_i)\}_{i=1}^{n_t}$ contains n_t images x_i and corresponding identity labels y_i . During testing, to verify the acquisition and anti-forgetting capacity of the method, the final model is evaluated on a series of T testing datasets $\mathcal{D}^{te} = \{D_t^{te}\}_{t=1}^T$ collected from all the seen domains. Besides, to verify the generalization capacity of the model, additional U datasets $\mathcal{D}^{un} = \{D_i^{un}\}_{i=1}^U$ from unseen domains are tested.

During the t -th training step, the trained model is named Θ_t^s which mainly contains the short-term new knowledge, and Θ_t denotes the inference model, which is generated by fusing Θ_t^s and an old model. Besides, we also define a long-term model Θ_t^l , which indicates a backward rectified model storing balanced knowledge of previous $t-1$ ($t \geq 2$) domains. The network of

each model contains a backbone and an identity classifier, e.g., the backbone and classifier of the model Θ_t^s are denoted as ψ_t^s and ϕ_t^s , respectively. Given an input image x , ψ_t^s converts x into a feature vector $v \in \mathbb{R}^d$, where d is the feature dimension. Then ϕ_t^s takes v as the input and generates identity predictions. $\mathbf{R}_t$, $\mathbf{R}_t^s$ and $\mathbf{R}_t^l$ are the inter-instance affinity matrix extracted by Θ_t , Θ_t^s and Θ_t^l , respectively. $\tilde{\mathbf{R}}_{t-1}$ denotes an affinity matrix generated by the rectification process. $\Delta \mathbf{R}_t$ is the absolute difference of two affinity matrices. The element $(\mathbf{R}_t)_{ij}$ denotes the i -th row, j -th column of the matrix $\mathbf{R}_t$, with the same indexing convention applied to the other matrices. s_p and s_n are the thresholds to evaluate the relation prediction correctness of person pairs with the same identities and different identities, separately. σ is the softmax function. δ_t is a scalar representing the extent of knowledge discrepancy between the new and old models.

B. Long Short-Term Knowledge Consolidation

In this section, we present a basic version of our LSTKC++, named Long Short-Term Knowledge Consolidation (LSTKC), which is introduced in our conference paper [11]. As shown in Fig. 3, LSTKC consists of two LReID models, a Rectification-based Short-Term Knowledge Transfer module $\mathcal{M}_r$ and an Estimation-based Long-Term Knowledge Consolidation module $\mathcal{M}_e$.

1) *New ReID Knowledge Learning*: The learnable parameters of ψ_t and ϕ_t in the new model Θ_t^s are optimized by a cross-entropy loss:

$$\mathcal{L}_{ID} = -y \log(\sigma(\phi_t(x; \psi_t))), \quad (1)$$

where y is the identity label of x .

Furthermore, to enhance the discriminative capacity of the model, a normalization-guided Triplet loss [55] is adopted:

$$\mathcal{L}_{Tri} = \log(1 + \exp(\|\bar{v}_a - \bar{v}_p\|_2^2 - \|\bar{v}_a - \bar{v}_n\|_2^2)), \quad (2)$$

where $\bar{v}$ is generated from v with L_2 -normalization, and $\langle \bar{v}_a, \bar{v}_p, \bar{v}_n \rangle$ is a triplet set [18]. Therefore, the overall loss of the new ReID knowledge learning process is:

$$\mathcal{L}_{Base} = \mathcal{L}_{ID} + \gamma \mathcal{L}_{Tri}, \quad (3)$$

where γ is a hyperparameter to balance two components.

In this paper, we adopt the training of a single model using the new data under the supervision of $\mathcal{L}_{Base}$ as the baseline, termed $\mathcal{M}_b$ in the subsequent sections.

2) *Rectification-based Short-Term Knowledge Transfer*: In this section, we focus on rectifying the erroneous knowledge of the old model and distilling the rectified knowledge to the new model. In the following, the process of *knowledge representation*, *short-term knowledge rectification*, and *rectified knowledge transfer* are introduced sequentially.

Knowledge Representation: A core function of LReID models is to assess the identity relations between different person images. In light of this, we introduce a pairwise relation matrix to effectively capture and represent the knowledge embedded within the LReID models. Specifically, during training step t ($t > 1$), given a batch of data $\mathcal{I} = \{(x_{t,i}, y_{t,i})\}_{i=1}^B$ where B is the batch size. Θ_t^s extracts a feature matrix $\mathbf{V}_t^s \in \mathbb{R}^{B \times d}$

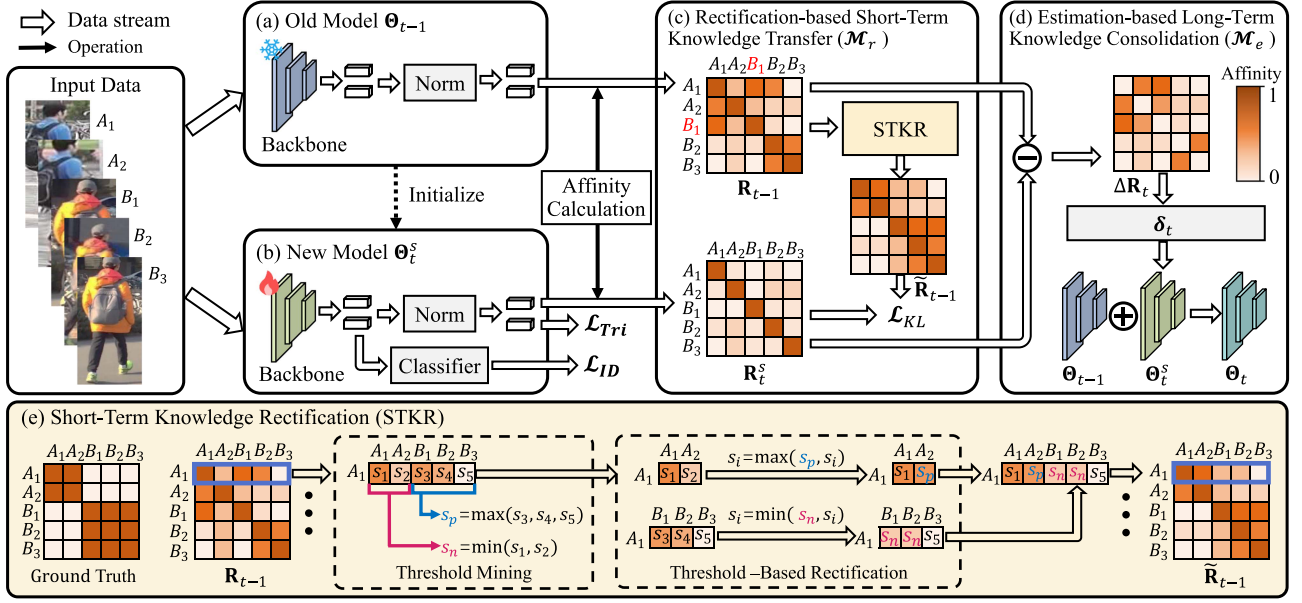


Fig. 3. The overall pipeline of our proposed LSTKC. (a) and (b) are the old and new models, respectively. (c) is the Rectification-based Short-Term Knowledge Transfer module that performs Short-Term Knowledge Rectification (STKR) to the affinity matrix predicted by the old model, and then distills the rectified knowledge to the new model. (d) is the Estimation-based Long-Term Knowledge Consolidation module that evaluates the knowledge difference between the new and old models to achieve a balance between new and old knowledge. (e) is the illustration of the Short-Term Knowledge Rectification procedure.

from $\mathcal{I}$. Then L_2 normalization is adopted to transform $\mathbf{V}_t^s$ into $\bar{\mathbf{V}}_t^s$. A pair-wise similarity matrix $\mathbf{S}_t^s \in \mathbb{R}^{B \times B}$ is calculated by $\mathbf{S}_t^s = \bar{\mathbf{V}}_t^s \cdot (\bar{\mathbf{V}}_t^s)^\top$, which is utilized to obtain the pair-wise relation matrix $\mathbf{R}_t^s \in \mathbb{R}^{B \times B}$ through affinity calculation:

$$(\mathbf{R}_t^s)_{ij} = \frac{\exp((\mathbf{S}_t^s)_{ij}/\tau)}{\sum_{k=1}^B \exp((\mathbf{S}_t^s)_{ik}/\tau)}, \quad (4)$$

where τ is a temperature hyper-parameter to scale the similarity scores. Equation (4) is a softmax-like function that maps each row of similarity scores into a relation score vector, which reflects the overall affinity between $x_{t,i}$ and other images of $\mathcal{I}$.

In addition, to represent the knowledge of the old model Θ_{t-1} , we feed the image batch $\mathcal{I}$ to Θ_{t-1} to obtain the pair-wise relation matrix $\mathbf{R}_{t-1}$ following the above process.

Short-Term Knowledge Rectification: The identification of erroneous predictions is based on two principles: (a) For a given instance, its affinity with instances of the same ID should be higher than the maximum affinity with instances of other IDs. (b) For a given instance, its affinity with instances of different IDs should be lower than the minimum affinity with instances of the same ID. These principles enable the detection of erroneous intra-ID relation scores and erroneous inter-ID relation scores, respectively.

To achieve this, we first propose a Threshold Mining strategy to support erroneous relation identification. As shown in Fig. 3 (c), for the i -th row in $\mathbf{R}_{t-1}$, we obtain the negative threshold s_n by:

$$s_n = \min\{(\mathbf{R}_{t-1})_{ij}\}_{j=1}^B \text{ w.r.t : } y_i = y_j. \quad (5)$$

Similarly, the positive threshold s_p is obtained by:

$$s_p = \max\{(\mathbf{R}_{t-1})_{ij}\}_{j=1}^B \text{ w.r.t : } y_i \neq y_j. \quad (6)$$

For example, in Fig. 3, instances A_1 and A_2 share the same ID, and B_1, B_2, B_3 belong to a different ID. Thus, the negative and positive thresholds for the first row are calculated as $s_n = \min\{s_1, s_2\}$ and $s_p = \max\{s_3, s_4, s_5\}$ respectively.

Next, we propose a Threshold-based Rectification strategy to correct erroneous relation scores. Specifically, the lower bound of intra-ID relation scores is set to s_p , and the upper bound of inter-ID relation scores is limited to s_n . Thus the rectified matrix $\tilde{\mathbf{R}}_{t-1}$ is then obtained by applying the following rules:

$$(\tilde{\mathbf{R}}_{t-1})_{ij} = \begin{cases} \max\{(\mathbf{R}_{t-1})_{ij}, s_p\} & \text{w.r.t : } y_i = y_j \\ \min\{(\mathbf{R}_{t-1})_{ij}, s_n\} & \text{w.r.t : } y_i \neq y_j \end{cases} \quad (7)$$

Then, to ensure the rows of the rectified relation matrix comply with the probability distribution, we apply L_1 normalization to each row of $\tilde{\mathbf{R}}_{t-1}$.

Rectified Knowledge Transfer: To distill the rectified old relation knowledge to the new model Θ_t , we adopt Kullback Leibler divergence (KL divergence), which is calculated by:

$$\mathcal{L}_{KL} = \frac{1}{B} \sum_{i=1}^B KL((\tilde{\mathbf{R}}_{t-1})_i || (\mathbf{R}_t)_i). \quad (8)$$

3) Estimation-based Long-Term Knowledge Consolidation: In this section, we aim to enhance knowledge consolidation by fusing the models of different stages. Instead of using fixed fusion parameters, we propose adaptively balancing the new and old knowledge based on the knowledge difference estimation of the new and old models.

Knowledge-guided Model Fusion: We use Θ_{t-1} and Θ_t^s to process all the training data, the resulting relation matrices are noted as $\mathbf{R}_{t-1}$ and $\mathbf{R}_t^s$, respectively. As is shown in Fig. 3 (d), we first compute the element-wise absolute difference between $\mathbf{R}_{t-1}$ and $\mathbf{R}_t^s$, denoted as $\Delta \mathbf{R}_t \in \mathbb{R}^{n_t \times n_t}$. This difference is

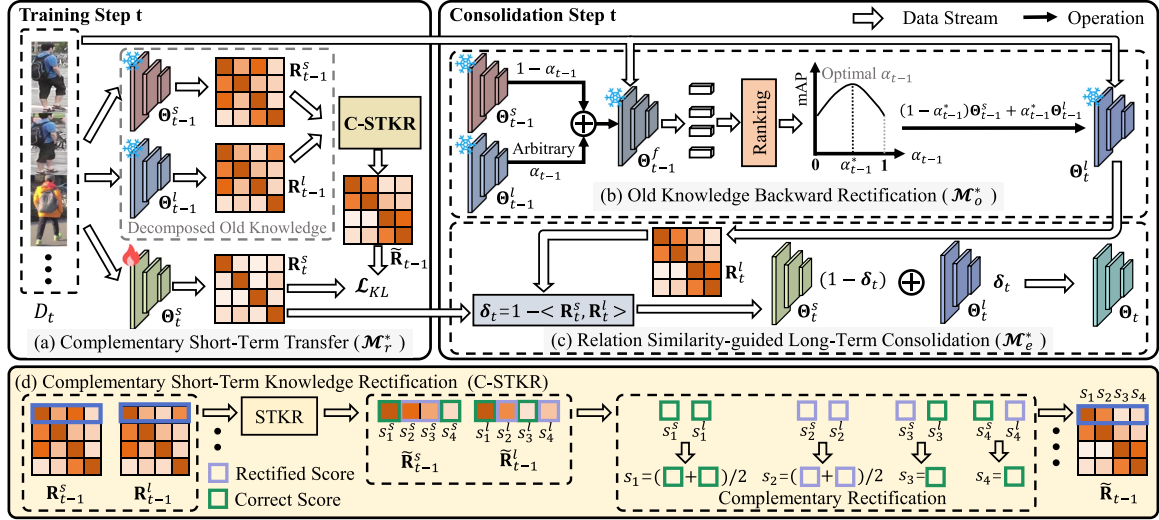


Fig. 4. The overall pipeline of our LSTKC++ framework. A short-term old model Θ_{t-1}^s and a long-term old model Θ_{t-1}^l are adopted. First, the Complementary Short-Term Transfer module transfers the complementary and rectified knowledge of Θ_{t-1}^s and Θ_{t-1}^l , which is obtained by the Complementary Short-Term Knowledge Rectification (C-STKR) strategy, to the new model Θ_t^s . Then, the Old Knowledge Backward Rectification module is adopted to search the optimal model fusion weight, resulting in an updated long-term model Θ_t^l . Finally, a Relation Similarity-guided Long-Term Consolidation module is developed to generate the model Θ_t for inference.

then summarized into a scalar δ_t by:

$$\delta_t = \frac{1}{n_t} \sum_{i=1}^{n_t} \left(\sum_{j=1}^{n_t} (\Delta \mathbf{R}_t)_{ij} \right), \quad (9)$$

where $\sum_{j=1}^{n_t} \Delta(\mathbf{R}_t)_{ij}$ evaluates the knowledge difference reflected on image $x_{t,i}$, and $\frac{1}{n_t} \sum_{i=1}^{n_t} (\cdot)$ calculates the average knowledge difference reflected by all images of D_t . Thus δ_t represents the overall knowledge discrepancy between Θ_{t-1} and Θ_t^s . Then, the final model Θ_t in step t is obtained by:

$$\Theta_t = (1 - \delta_t) \Theta_t^s + \delta_t \Theta_{t-1}, \quad (10)$$

where δ_t serves as a weight balancing the short-term new knowledge and long-term historical knowledge.

4) *Training and Inference*: For the first training step, $\mathcal{L}_{Base}$ is directly adopted as the loss to obtain the model Θ_1 . For the subsequent steps, the old model Θ_{t-1} and the new data D_t are utilized to learn Θ_t^s , the total training loss function is formulated as

$$\mathcal{L} = \mathcal{L}_{Base} + \lambda \mathcal{L}_{KL}, \quad (11)$$

where λ is a weight to balance the plasticity and stability of the Θ_t^s . When the training is finished, we adopt $\mathcal{M}_e$ to fuse Θ_{t-1} and Θ_t^s to derive the output model Θ_t .

During inference, the final model Θ_T is used to extract image features for person ranking.

C. Long Short-Term Knowledge Decomposition and Consolidation

In this section, we introduce the Long Short-Term Knowledge Decomposition and Consolidation method, LSTKC++, which builds upon the LSTKC framework with the same backbone and ReID knowledge learning loss. Consequently, in Fig. 4, we omit the ReID loss and the affinity calculation process for clarity. As

shown in Fig. 4, instead of maintaining a single old model Θ_{t-1} , we adopt a long-term old model Θ_{t-1}^l and a short-term old model Θ_{t-1}^s . To fully exploit the knowledge from both models, enhance the balance of old knowledge, and improve the robustness of the final model, we introduce the following three modules:

1) *Complementary Short-Term Transfer*: Complementary Short-Term Transfer module ($\mathcal{M}_r^*$) extends the previously introduced $\mathcal{M}_r$ module. As shown in Fig. 4(a), during the new model training, Θ_{t-1}^s and Θ_{t-1}^l are leveraged to generate the relation matrix $\mathbf{R}_{t-1}^s$ and $\mathbf{R}_{t-1}^l$ respectively. Then, a Complementary Short-Term Knowledge Rectification (C-STKR) strategy is developed to utilize the complementary knowledge encoded in $\mathbf{R}_{t-1}^s$ and $\mathbf{R}_{t-1}^l$ to improve knowledge rectification.

Specifically, we utilize the STKR strategy described in (Section III-B) to rectify the $\mathbf{R}_{t-1}^s$ and $\mathbf{R}_{t-1}^l$, resulting in $\tilde{\mathbf{R}}_{t-1}^s$ and $\tilde{\mathbf{R}}_{t-1}^l$. In Fig. 4(d), the originally erroneous relation scores, which are rectified by the STKR module, are highlighted in light purple, whereas the correct relation scores that remain unchanged are marked in green. For each element in $\tilde{\mathbf{R}}_{t-1}^s$ and $\tilde{\mathbf{R}}_{t-1}^l$, the following complementary rectification strategy is implemented: (1) When both models provide correct or incorrect relation scores, $[(\tilde{\mathbf{R}}_{t-1}^s)_{ij} + (\tilde{\mathbf{R}}_{t-1}^l)_{ij}]/2$ is preserved. (2) If one model predicts the relation score correctly and the other incorrectly, only the correct score is retained. The resulting rectified matrix, $\tilde{\mathbf{R}}_{t-1}$, is used to compute the knowledge transfer loss, as defined in (8). Besides, the overall loss of LSTKC++ follows the same formulation as in (11).

2) *Old Knowledge Backward Rectification*: The Old Knowledge Backward Rectification module ($\mathcal{M}_o^*$) aims to obtain a rectified fusion weight of the old models to achieve better acquisition and forgetting balance. The fusion process under an arbitrary parameter α_{t-1} can be written as

$$\Theta_t^f = (1 - \alpha_{t-1}) \Theta_{t-1}^s + \alpha_{t-1} \Theta_{t-1}^l, \quad (12)$$

Algorithm 1: Old Knowledge Backward Rectification ($\mathcal{M}_o^*$).

Input: Training data $D_t = \{x_i, y_i\}_{i=1}^{n_t}$. Long-term old model Θ_{t-1}^l and short-term old model Θ_{t-1}^s which are provided by step $t-1$.
Output: Long-Term old model Θ_t^l of step t .

- 1: best_mAP=0;
- 2: $\alpha_{t-1}^* = 0$;
- 3: **for** $\alpha_{t-1} \leftarrow 0$ to 1 interval 0.1 **do**
- 4: $\Theta_{t-1}^f = (1 - \alpha_{t-1})\Theta_{t-1}^s + \alpha_{t-1}\Theta_{t-1}^l$, (12);
- 5: Extract features $\mathcal{F} = \{f_i\}_{i=1}^{n_t}$ of $\{x_i\}_{i=1}^{n_t}$ with Θ_{t-1}^f ;
- 6: Taking $\mathcal{F}$ as the query and gallery to execute ranking;
- 7: Calculate ranking mAP_t according to $\mathcal{Y} = \{y_i\}_{i=1}^{n_t}$;
- 8: **if** mAP_t > best_mAP **then**
- 9: $\alpha_{t-1}^* = \alpha_{t-1}$;
- 10: best_mAP = mAP_t;
- 11: **end if**
- 12: **end for**
- 13: $\Theta_t^l = (1 - \alpha_{t-1}^*)\Theta_{t-1}^s + \alpha_{t-1}^*\Theta_{t-1}^l$, (14)

where Θ_t^f is the fused model. Since the new data D_t is unseen during Θ_{t-1}^l and Θ_{t-1}^s training, it can serve as an unbiased reference to evaluate the fused models. Therefore, the optimal weight α_{t-1}^* can be obtained by the following function:

$$\alpha_{t-1}^* = \arg \max \text{Ranking}(\Theta_{t-1}^f, D_t), \quad (13)$$

where $\text{Ranking}(\Theta_{t-1}^f, D_t)$ denotes using Θ_{t-1}^f to extract the features of instances in D_t and computing the ranking performance. In this paper, we adopt the mAP [18] as the ranking metric. Practically, we obtain α_{t-1}^* with a simple linear search. Finally, the updated long-term knowledge model is accomplished by:

$$\Theta_t^l = (1 - \alpha_{t-1}^*)\Theta_{t-1}^s + \alpha_{t-1}^*\Theta_{t-1}^l. \quad (14)$$

The overall process of the $\mathcal{M}_o^*$ module is shown in Algorithm 1.

3) *Relation Similarity-guided Long-Term Consolidation*: Relation Similarity-guided Long-Term Consolidation ($\mathcal{M}_e^*$) is an extension of the $\mathcal{M}_e$ module, aiming to achieve a superior new-old model fusion weight prediction. Although (9) provides a solution to generate model fusion weight, we theoretically (Section IV) prove that (9) is sensitive to the domain gaps between datasets. As the domain gap between new and old datasets is large, (9) can generate a fusion weight exceeding 1.0, which leads to the degraded performance of the fused model. Therefore, we propose an improved strategy to obtain δ_t .

In the previous section, $\mathcal{M}_o^*$ provides a long-term old knowledge model Θ_t^l containing abundant knowledge of previous $t-1$ domains. To predict the appropriate weight to fuse Θ_t^l and Θ_t^s , we first obtain the relation matrices $\mathbf{R}_t^l \in \mathbb{R}^{n_t \times n_t}$ and $\mathbf{R}_t^s \in \mathbb{R}^{n_t \times n_t}$ generated by Θ_t^l and Θ_t^s . Then, the old model

Algorithm 2: Training Process of LSTKC++.

Input: Training data stream $\mathcal{D} = \{D_t\}_{t=1}^T$. ImageNet pretrained model Θ_0 .
Output: Inference model of each training step Θ_t .

- 1: **for** $t \leftarrow 1$ to T **do**
- 2: **if** $t=1$ **then**
- 3: $\Theta_1^s = \mathcal{M}_b(\Theta_0, D_t)$, Fig. 3(b) and (3);
- 4: $\Theta_1 = \Theta_1^s$;
- 5: **else if** $t=2$ **then**
- 6: $\Theta_2^s = \mathcal{M}_r(\Theta_1, D_t)$, Fig. 3(c);
- 7: $\Theta_2^l = \Theta_1^s$;
- 8: $\Theta_2 = \mathcal{M}_e^*(\Theta_2^l, \Theta_2^s, D_t)$, Fig. 4(c);
- 9: **else**
- 10: $\Theta_t^s = \mathcal{M}_r^*(\Theta_{t-1}^l, \Theta_{t-1}^s, D_t)$, Fig. 4(a);
- 11: $\Theta_t^l = \mathcal{M}_o^*(\Theta_{t-1}^l, \Theta_{t-1}^s)$, Fig. 4(b);
- 12: $\Theta_t = \mathcal{M}_e^*(\Theta_t^l, \Theta_t^s, D_t)$, Fig. 4(c);
- 13: **end if**
- 14: **end for**

weight is calculated by:

$$\delta_t = 1 - \frac{1}{n_t} \sum_{i=1}^{n_t} \langle (\mathbf{R}_t^s)_i, (\mathbf{R}_t^l)_i \rangle, \quad (15)$$

where $\langle (\mathbf{R}_t^s)_i, (\mathbf{R}_t^l)_i \rangle$ denotes the dot product between the vectors $(\mathbf{R}_t)_i$ and $(\mathbf{R}_t^l)_i$, reflecting the similarity of cross-instance relations predicted by Θ_t^s and Θ_t^l . Finally, the model for inference, i.e., Θ_t , is obtained by

$$\Theta_t = (1 - \delta_t)\Theta_t^s + \delta_t\Theta_t^l. \quad (16)$$

4) *Training and Inference*: For LSTKC++, when $t = 1$, the model is trained with $\mathcal{L}_{Base}$ directly, i.e., utilizing $\mathcal{M}_b$. When $t = 2$, the same knowledge rectification strategy and loss function from LSTKC are applied, and Θ_2^l share the same parameters with Θ_1^s . When $t \geq 2$, the model updates with the pipeline shown in Fig. 4. The overall training process of LSTKC++ is visualized in Algorithm 2. During inference, the final model Θ_T is adopted to extract human features.

IV. THEORETICAL ANALYSES AND JUSTIFICATIONS

Q1: What is the influence of model fusion on knowledge consolidation in lifelong learning?

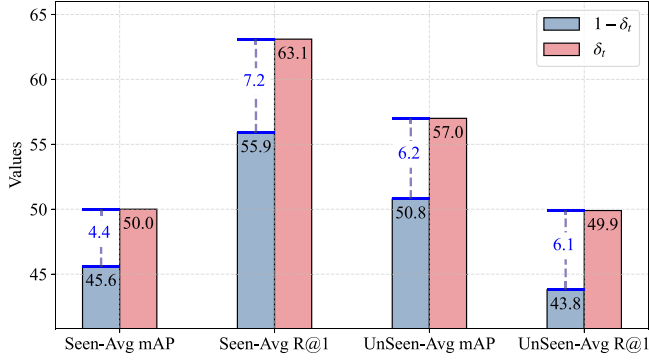
After the lifelong learning procedure with T steps, according to (10), the final model Θ_T could be represented as:

$$\begin{aligned} \Theta_T &= (1 - \delta_T)\Theta_T^s + \delta_T((1 - \delta_{T-1})\Theta_{T-1}^s + \dots) \\ &= \beta_T\Theta_T^s + \beta_{T-1}\Theta_{T-1}^s + \dots + \beta_1\Theta_1^s, \end{aligned} \quad (17)$$

where $\beta_t = (1 - \delta_t) \prod_{i=t+1}^T \delta_i$, and $\delta_1 = 0$ (because Θ_1 is directly learned without fusion process). It is evident that the final model is equivalent to a weighted fusion of all previous models that assures long-term knowledge accumulation.

Q2: Why δ_t is used as the old model weight instead of $1 - \delta_t$?

Taking the LSTKC framework as the example, at learning step t , the old model Θ_{t-1} retains knowledge from previous $t-1$ steps, while the new model Θ_t^s primarily focuses on the t -th step. To mitigate catastrophic forgetting on the historical


 Fig. 5. Performance with δ_t and $1 - \delta_t$ as old model weight.

$t-1$ steps which is the majority of learned knowledge, a larger weight δ_t is assigned to Θ_{t-1} when the knowledge difference is more significant as in (10), thereby achieving a better balance between acquisition and forgetting. To verify this, we conducted two experiments that exploit δ_t and $1 - \delta_t$ as the weights of the old model Θ_{t-1} separately. The results in Fig. 5 demonstrate that using δ_t as the old model weight achieves significantly superior overall performance than $1 - \delta_t$.

Q3: The difference between model knowledge difference calculation strategies (9) vs. (15)

Since each row $\mathbf{r} \in \mathbb{R}^{n_t}$ of the relation matrices is generated through a softmax-like function, $\mathbf{r}$ is a non-negative vector that satisfies:

$$\begin{cases} \sum_{j=1}^{n_t} \mathbf{r}_j = 1 \\ 0 \leq \mathbf{r}_j \leq 1 \end{cases}, \quad (18)$$

where $\mathbf{r}_j$ is the j -th element of $\mathbf{r}$.

Denote $d_e(\mathbf{r}^{(1)}, \mathbf{r}^{(2)})$ is the absolute difference between two relation vectors $\mathbf{r}^{(1)}$ and $\mathbf{r}^{(2)}$, which can be derived as follows:

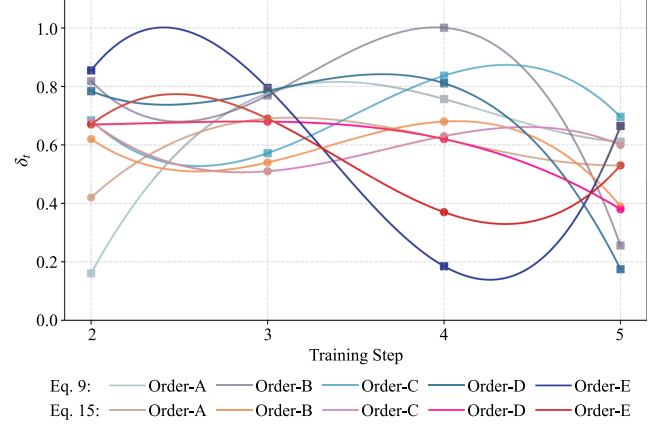
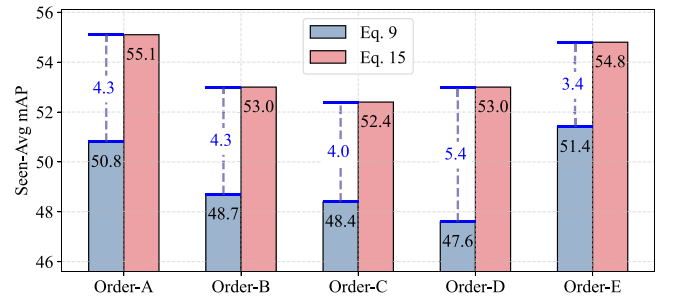
$$\begin{aligned} d_e(\mathbf{r}^{(1)}, \mathbf{r}^{(2)}) &= \sum_{i=1}^{n_t} |\mathbf{r}_i^{(1)} - \mathbf{r}_i^{(2)}| \\ &\leq \sum_{i=1}^{n_t} (|\mathbf{r}_i^{(1)}| + |\mathbf{r}_i^{(2)}|) = \sum_{i=1}^{n_t} \mathbf{r}_i^{(1)} + \sum_{i=1}^{n_t} \mathbf{r}_i^{(2)} = 2. \end{aligned} \quad (19)$$

Since (9) calculates the average absolute difference across different rows of $\mathbf{R}_t^s$ and $\mathbf{R}_t^l$ which could be written as $\delta_t = \frac{1}{n_t} \sum_{i=1}^{n_t} d_e((\mathbf{R}_t^s)_i - (\mathbf{R}_t^l)_i) \leq 2$. Thus, δ_t can be higher than 1 when the domain gap between the new and old dataset is significant, leading to a negative weight to the new model.

Besides, denote $d_s(\mathbf{r}^{(1)}, \mathbf{r}^{(2)})$ is the similarity between two relation vectors $\mathbf{r}^{(1)}$ and $\mathbf{r}^{(2)}$, which satisfies:

$$\begin{cases} d_s(\mathbf{r}^{(1)}, \mathbf{r}^{(2)}) = \sum_{i=1}^{n_t} \mathbf{r}_i^{(1)} \mathbf{r}_i^{(2)} \leq \sum_{i=1}^{n_t} \mathbf{r}_i^{(1)} \sum_{i=1}^{n_t} \mathbf{r}_i^{(2)} = 1 \\ d_s(\mathbf{r}^{(1)}, \mathbf{r}^{(2)}) = \sum_{i=1}^{n_t} \mathbf{r}_i^{(1)} \mathbf{r}_i^{(2)} \geq \sum_{i=1}^{n_t} 0 = 0 \end{cases}. \quad (20)$$

In (15), the average similarity difference across different rows of $\mathbf{R}_t^s$ and $\mathbf{R}_t^l$ is calculated, which can be expressed as $\delta_t = 1 - \frac{1}{n_t} \sum_{i=1}^{n_t} d_s((\mathbf{R}_t^s)_i - (\mathbf{R}_t^l)_i) \in [0, 1]$. This formulation ensures that δ_t derived from (15) prevents sign inversion in Θ_t^s during model fusion. Consequently, it enhances the robustness


 Fig. 6. The δ_t predicted by (9) and (15) in LSTKC++. Five different dataset orders are adopted.

 Fig. 7. Performance of different δ_t calculation strategies.

of the fusion model when facing large inter-step domain gaps, performing more reliably than (9).

We visualize the δ_t predicted by (9) and (15) in Fig. 6, where we test five different dataset orders and calculate δ_t starting from the second step in each training order. The visualized results confirm that the predictions from (15) (red curves) are more stable. Additionally, Fig. 7 demonstrates consistent LReID performance improvement when substituting (9) with (15). This improvement is attributed to the large domain gap between different datasets. Besides, due to the LSTKC++ framework's use of two old models, which introduce more diverse knowledge and result in a larger knowledge gap between the new model Θ_t^s and fused long-term model Θ_t^l in some conditions. Hence, a more stable δ_t prediction strategy is more effective.

V. EXPERIMENTS

A. Datasets and Evaluation Metrics

Datasets: We conducted all experiments on the widely-used LReID benchmark [4], which consists of 12 datasets. Among them, five datasets (Market-1501 [1], DukeMTMC [56], CUHK-SYSU [57], MSMT17-V2 [16], and CUHK03 [58]) are adopted as seen domains for LReID training and anti-forgetting evaluation. The remaining seven datasets (CUHK01 [59], CUHK02 [60], VIPeR [61], PRID [62], i-LIDS [63], GRID [64], and SenseReID [65]) are unseen during training and only used for generalization evaluation. Note that the LReID benchmark

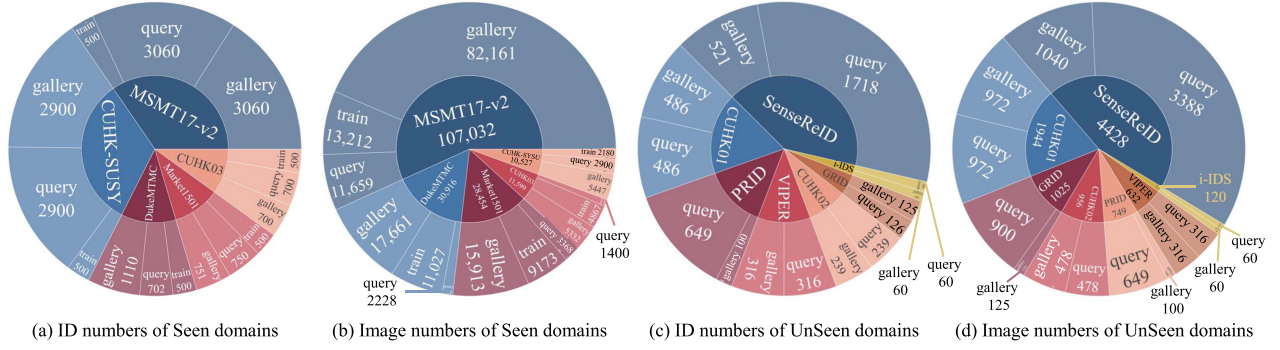


Fig. 8. ID and image number statistics of seen and unseen domains in LReID benchmark.

selects 500 identities from each seen dataset for training. Typically, two training orders are adopted to simulate different LReID scenarios^{1,2}. The statistical information of all the above datasets is shown in Fig. 8. For each dataset, the “train” subset is used for lifelong training, and the “query” and “gallery” subsets are only used for evaluation.

Evaluation Metrics: Following the evaluation settings of the existing LReID methods [4], [10], we calculate mean Average Precision (mAP) and Rank@1 (R@1) accuracy on each dataset and use the average of mAP and R@1 results on the seen and unseen datasets to evaluate the overall performance of the LReID models.

B. Implementation Details

Following the previous LReID works [4], [10], we adopt ResNet-50 [66] as our backbone. For both training orders, the first dataset is trained for 80 epochs and the subsequent datasets are trained for 60 epochs. The SGD optimizer with a momentum of 0.9 is used for model optimization. The learning rate is set to 8×10^{-3} initially with 0.1 decay at the 30th epoch. The input images are resized to 256×128 with random cropping, erasing, and horizontal flipping augmentation. The batch size is set to 64, with 8 identities and 8 images for each identity. The hyperparameters γ , τ , and λ are set to 1, 0.1, and 1, respectively. Our implementation is based on PyTorch. All experiments are conducted on a single NVIDIA 4090 GPU. In the $\mathcal{M}_o$ module of LSTKC++, the linear search is conducted by sampling from 0 to 1 with an interval of 0.1.

C. Comparison Methods

For a thorough evaluation, we compare the proposed LSTKC and LSTKC++ with two kinds of state-of-the-art methods: *Class Incremental Learning (CIL)* methods (LwF [50], SPD [67], PRD [68], PRAKA [69], and FCS [70]) and *Non-exemplar LReID* approaches (CRL [71], AKA [4], PatchKD [10], ConRFL [41], MEGE [72], CKP [44] and DKP [39]). Because different persons can be regarded as different classes, existing

state-of-the-art non-exemplar CIL methods can be readily compared in the task of LReID. Besides, the *JointTrain*, which indicates aggregating all available training data to jointly train the model, is also reported. Usually, *JointTrain* results are regarded as the upper-bound performance of LReID models [72]. For a fair comparison, all compared experimental results follow the official publications or are implemented with the official codes.

D. Seen-Domain Non-Forgetting Evaluation

The performance of our LSTKC/LSTKC++ and the existing methods on the seen domains are reported in Tables I and II, following dataset order-1 and order-2, separately. We also calculate the average performance on all seen domains (Seen-Avg) in the tables. The best and second best results are marked in **RED** and **BLUE**, respectively.

Compared to CIL Methods: As is shown in Tables I and II, our LSTKC and LSTKC++ outperform all CIL methods with at least **4.6%/6.1%** Seen-Avg mAP/R@1 improvement. Especially, our LSTKC++ achieves the best performance on all datasets except for CUHK03 under both training orders. These superior results underscore the effectiveness of our LSTKC and LSTKC++ in addressing the LReID problem. This is because our models focus on consolidating the inter-instance relation knowledge, which is the essence of ReID, while the CIL methods focus on handling the semantic drift across learning steps. Due to semantic drift not necessarily affecting person-matching results, some constraints used in CIL not only fail to effectively improve the LReID performance but also limit the new knowledge-learning capacity of the model. Among the CIL methods, PRAKA outperforms our LSTKC and LSTKC++ on the last training dataset CUHK03. This is because PRAKA adopts soft constraints on the new model. As a result, PRAKA achieves better performance on the last dataset but experiences significant forgetting on the historical datasets. In contrast, our LSTKC and LSTKC++ models better consolidate inter-instance discriminative knowledge due to the relation-based knowledge rectification and transfer design.

Compared to LReID Methods: In Tables I and II, our LSTKC++ outperforms state-of-the-art DKP by **3.4%/2.6%** and **2.7%/1.5%** Seen-Avg mAP/R@1 on training order-1 and order-2, respectively. Considering DKP adopts about twice more parameters and GPU memory cost compared to our LSTKC++ as shown in Table V, our LSTKC++ is more **effective** and **efficient**.

¹(Order-1) Market-1501→CUHK-SYSU → DukeMTMC→MSMT17→CUHK03

²(Order-2) DukeMTMC → MSMT17→Market-1501→CUHK-SYSU → CUHK03

TABLE I
TRAINING ORDER-1: MARKET-1501 → CUHK-SYSU → DUKEMTMC → MSMT17 → CUHK03

	Method	Publication	Market-1501		CUHK-SYSU		DukeMTMC		MSMT17		CUHK03		Seen-Avg		UnSeen-Avg	
			mAP	R@1	mAP	R@1	mAP	R@1	mAP	R@1	mAP	R@1	mAP	R@1	mAP	R@1
CIL	JointTrain	-	68.1	85.2	81.4	83.8	60.4	75.7	24.6	48.9	42.7	43.6	55.4	67.5	49.8	46.3
	LwF [50]	T-PAMI 2017	56.3	77.1	72.9	75.1	29.6	46.5	6.0	16.6	36.1	37.5	40.2	50.6	47.2	42.6
	SPD [67]	ICCV 2019	35.6	61.2	61.7	64.0	27.5	47.1	5.2	15.5	42.2	44.3	34.4	46.4	40.4	36.6
	PRD [68]	ICML 2023	7.3	18.0	33.5	35.6	3.7	7.6	0.8	2.4	33.8	33.8	15.8	19.5	23.0	17.7
	PRAKA [69]	ICCV 2023	37.4	61.3	69.3	71.8	35.4	55.0	10.7	27.2	54.0	55.6	41.3	54.2	47.7	41.6
	FCS [70]	CVPR 2024	58.3	78.5	75.1	76.2	42.6	59.8	10.2	24.3	35.3	34.9	44.3	54.7	52.1	44.2
LReID	CRL [71]	WACV 2021	58.0	78.2	72.5	75.1	28.3	45.2	6.0	15.8	37.4	39.8	40.5	50.8	47.8	43.5
	AKA [4]	CVPR 2021	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† [4]	CVPR 2021	58.1	77.4	72.5	74.8	28.7	45.2	6.1	16.2	38.7	40.4	40.8	50.8	47.6	42.6
	PatchKD [10]	ACM MM 2022	68.5	85.7	75.6	78.6	33.8	50.4	6.5	17.0	34.1	36.8	43.7	53.7	49.1	45.4
	ConRFL [41]	PR 2023	59.2	78.3	82.1	84.3	45.6	61.8	12.6	30.4	51.7	53.8	50.2	61.7	57.4	52.3
	MEGE [72]	T-PAMI 2023	39.0	61.6	73.3	76.6	16.9	30.3	4.6	13.4	36.4	37.1	34.0	43.8	47.7	44.0
	CKP [44]	ACM MM 2024	53.8	76.0	81.2	83.0	49.7	67.0	18.4	40.8	44.1	45.8	49.4	62.5	58.0	51.0
	DKP [39]	CVPR 2024	60.3	80.6	83.6	85.4	51.6	68.4	19.7	41.8	43.6	44.2	51.8	64.1	59.2	51.6
	LSTKC	This Paper	54.7	76.0	81.1	83.4	49.4	66.2	20.0	43.2	44.7	46.5	50.0	63.1	57.0	49.9
	LSTKC++	This Paper	65.3	83.2	85.8	87.5	56.0	70.5	21.0	43.5	47.7	48.9	55.2	66.7	63.2	56.3

† indicates the results are re-implemented by [10].

TABLE II
TRAINING ORDER-2: DUKEMTMC → MSMT17 → MARKET-1501 → CUHK-SYSU → CUHK03

	Method	Publication	DukeMTMC		MSMT17		Market-1501		CUHK-SYSU		CUHK03		Seen-Avg		UnSeen-Avg	
			mAP	R@1	mAP	R@1	mAP	R@1	mAP	R@1	mAP	R@1	mAP	R@1	mAP	R@1
CIL	JointTrain	-	60.4	75.7	24.6	48.9	68.1	85.2	81.4	83.8	42.7	43.6	55.4	67.5	49.8	46.3
	LwF [50]	T-PAMI 2017	42.7	61.7	5.1	14.3	34.4	58.6	69.9	73.0	34.1	34.1	37.2	48.4	44.0	40.1
	SPD [67]	ICCV 2019	28.5	48.5	3.7	11.5	32.3	57.4	62.1	65.0	43.0	45.2	33.9	45.5	39.8	36.3
	PRD [68]	ICML 2023	3.6	8.2	0.6	1.8	8.9	22.3	34.6	36.1	35.4	35.3	16.6	20.7	20.7	15.0
	PRAKA [69]	ICCV 2023	31.2	48.7	6.6	19.1	47.8	69.8	70.4	73.0	54.9	56.6	42.2	53.4	48.4	41.1
	FCS [70]	CVPR 2024	53.6	70.0	9.5	23.5	48.7	69.9	76.2	78.2	37.1	38.4	45.0	56.0	52.7	45.1
LReID	CRL [71]	WACV 2021	43.5	63.1	4.8	13.7	35.0	59.8	70.0	72.8	34.5	36.8	37.6	49.2	45.3	41.4
	AKA [4]	CVPR 2021	32.5	49.7	-	-	-	-	-	-	-	-	-	-	40.8	37.2
	AKA† [4]	CVPR 2021	42.2	60.1	5.4	15.1	37.2	59.8	71.2	73.9	36.9	37.9	38.6	49.4	46.0	41.7
	PatchKD [10]	ACM MM 2022	58.3	74.1	6.4	17.4	43.2	67.4	74.5	76.9	33.7	34.8	43.2	54.1	48.6	44.1
	ConRFL [41]	PR 2023	34.4	51.3	7.6	20.1	61.6	80.4	82.8	85.1	49.0	50.1	47.1	57.4	57.9	53.4
	MEGE [72]	T-PAMI 2023	21.6	35.5	3.0	9.3	25.0	49.8	69.9	73.1	34.7	35.1	30.8	40.6	44.3	41.1
	CKP	ACM MM 2024	49.4	67.0	14.5	33.8	56.0	77.6	83.2	84.9	45.3	47.1	49.7	62.1	57.2	50.0
	DKP [39]	CVPR 2024	53.4	70.5	14.5	33.3	60.6	81.0	83.0	84.9	45.0	46.1	51.3	63.2	59.0	51.6
	LSTKC	This Paper	49.9	67.6	14.6	34.0	55.1	76.7	82.3	83.8	46.3	48.1	49.6	62.1	57.6	49.6
	LSTKC++	This Paper	55.3	70.6	15.7	34.5	65.6	82.7	86.3	88.0	47.1	47.6	54.0	64.7	62.1	54.7

† indicates the results are re-implemented by [10].

TABLE III
ABLATION STUDY ON COMPONENTS OF LSTKC AND LSTKC++

Method	$\mathcal{M}_b$	$\mathcal{M}_r$	$\mathcal{M}_e$	Seen-Avg		UnSeen-Avg	
				mAP	R@1	mAP	R@1
Baseline	✓			40.4	53.3	48.2	41.4
	✓	✓		46.1	59.7	54.2	47.9
	✓		✓	46.3	59.5	53.6	46.2
	✓	✓	✓	50.0	63.1	57.0	49.9
LSTKC	$\mathcal{M}_b$	$\mathcal{M}_r^*$	$\mathcal{M}_e^*$				
	✓	✓		46.5	58.9	55.3	49.0
	✓		✓	48.3	57.3	55.8	48.1
	✓		✓	52.5	64.0	59.3	51.3
LSTKC++	✓	✓	✓	55.2	66.7	63.2	56.3

TABLE IV
ABLATION ON CHOICES OF MINI-BATCH

Batch Size	Num-IDs	Num-Imgs	Seen-Avg		UnSeen-Avg	
			mAP	R@1	mAP	R@1
32	4	8	54.4	66.5	63.7	56.6
64	4	16	54.5	66.6	64.0	56.7
64	8	8	55.2	66.7	63.2	56.3
64	16	4	54.0	65.8	61.2	54.3
128	8	16	54.9	66.6	62.9	55.6
128	16	8	53.4	65.7	59.4	52.6
128	32	4	53.0	65.2	60.0	52.3
256	16	16	52.4	64.5	60.2	53.7
256	32	8	51.2	63.1	58.4	51.5
256	64	4	50.3	62.7	56.3	48.7

TABLE V
COMPUTING EFFICIENCY COMPARISON WITH EXISTING LReID METHODS

Method	Batch Time ↓ (s)	Params ↓ (M)	Model Memory ↓ (MB)	GPU Memory ↓ (MB)
PatchKD	0.303	42.20	161.51	12984
AKA	0.325	32.98	126.71	12946
DKP	0.716	97.89	389.42	33518
LSTKC	0.272	28.65	109.55	10998
LSTKC++	0.356	28.65	219.10	11816

Compared to the other LReID methods, our LSTKC++ achieves at least **5.0%/5.0%** and **6.9%/7.3%** Seen-Avg mAP/R@1 improvement on training order-1 and order-2.

On each individual dataset, compared to existing LReID methods, our LSTKC++ models consistently achieve superior or competitive performance. We do observe inferior results are obtained when compared to PatchKD and ConRFL on the first and last datasets, respectively. The former is attributed to the strict knowledge distillation loss used in PatchKD, which forces the network to maintain the learned knowledge on the first dataset. However, the erroneous knowledge of the first dataset also hinders the new knowledge acquisition capacity of the model, leading to degraded performance in sequential datasets. Compared to ConRFL, since our method comprises a balance

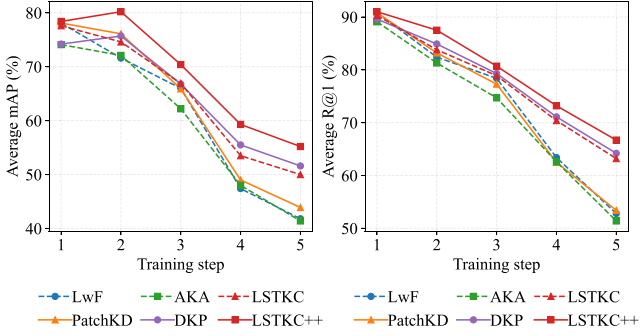


Fig. 9. Illustration of seen domain knowledge consolidation along the training process under training order-1.

between the new and old knowledge, the marginal performance loss on the last dataset leads to significant improvement in the performance of the middle datasets.

Seen Domain Performance Tendency: To comprehensively show the anti-forgetting capacity of our method compared to existing approaches, we test the models learned at the t -th step on seen domains. Fig. 9 visualizes the trends of the average mAP and R@1 of each method. As the learned domain increases, the performance of the existing methods LwF, AKA, and PatchKD drops rapidly, while our LSTKC outperforms them significantly after the 4-th step, showcasing the knowledge consolidation capacity of our model. Besides, our LSTKC++ consistently outperforms the state-of-the-art DKP method due to the long-term and short-term decomposed knowledge utilization mechanism.

E. Unseen-Domain Generalising Evaluation

Performance Analysis: The average performance on unseen datasets is shown in the “UnSeen-Avg” items of Tables I and II. The results show that our LSTKC++ achieves superior UnSeen-Avg mAP/R@1 performance compared to all existing CIL and LReID methods. Specifically, at least **4.0%/4.0%** and **3.1%/1.3%** UnSeen-Avg mAP/R@1 improvement is obtained compared to the state-of-the-art methods. The effectiveness of our model is attributed to the $\mathcal{M}_o^*$ module, which continually utilizes the unseen new data to determine the appropriate fusion weight of the old models.

Unseen Domain Generalization Tendency: To show how different methods accumulate generalization knowledge along the lifelong training process, we test the models on all unseen datasets after each training step. The results are illustrated in Fig. 10. In the first step, our LSTKC/LSTKC++ and state-of-the-art DKP outperform the other LReID methods by a minor margin because no additional constraints are applied to the model learning. As the number of training steps increases, the performance gap progressively widens, indicating the superior ability of our LSTKC/LSTKC++ in capturing and consolidating generalizable knowledge compared to LwF, AKA, and PatchKD. Compared to the state-of-the-art DKP, our LSTKC++ also achieves increasing improvement, showcasing the superiority of our method in continually consolidating generalizable knowledge.

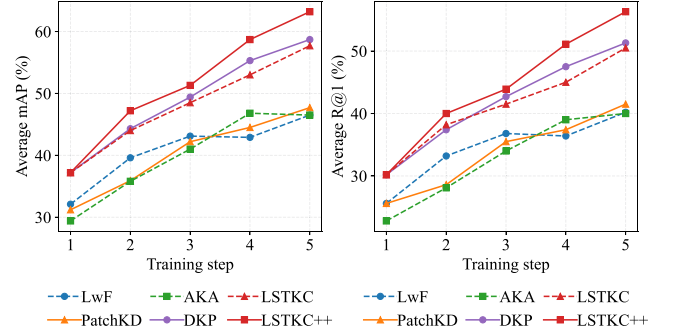


Fig. 10. Illustration of unseen datasets generalization along the training process under training order-1.

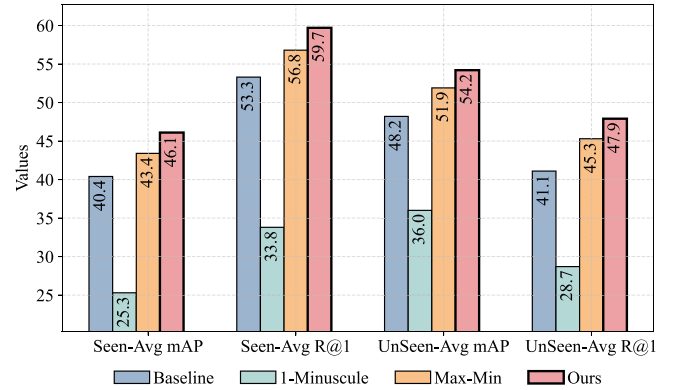


Fig. 11. Effectiveness of different rectification strategies.

F. Ablation Studies

1) Ablation Studies on LSTKC

The effectiveness of LSTKC components: In Table III, we present the ablation studies of LSTKC components, including $\mathcal{M}_r$ and $\mathcal{M}_e$. We start from a baseline model that merely uses $\mathcal{L}_{base}$ as the loss function and adds $\mathcal{M}_r/\mathcal{M}_e$ gradually. From the second row and the third row, we can observe that $\mathcal{M}_r$ and $\mathcal{M}_e$ could improve mAP/R@1 by about 4.8%-6.2% over baseline when used alone. When they are used together, the improvement is up to 8.5%-9.8%. The results show that both $\mathcal{M}_r$ and $\mathcal{M}_e$ can work individually and cooperatively.

Ablation on knowledge rectification strategy in $\mathcal{M}_r$: Knowledge rectification plays a vital role in $\mathcal{M}_r$. Apart from the introduced strategy in Fig. 3, we have investigated alternative knowledge rectification strategies. For simplicity, we denote the identical-ID pairs with similarity scores lower than s_p as FN and the non-identical pairs with similarity higher than s_n as FP. The relevant strategies include replacing FN and FP similarity scores with (a) the maximum and minimum similarity scores of person pairs (noted as Max-Min), (b) 1 and minuscule value (noted as 1-Minuscule) respectively, where minuscule value is set to 10^{-5} to avoid the calculation error in KL divergence loss. The results are shown in Fig. 11. The proposed method in this paper outperforms Max-Min and 1-Minuscule significantly. This is because the change of specific affinity involves the entire affinity distribution alteration within each row of the relation Matrix, where excessive modification would destroy the correct

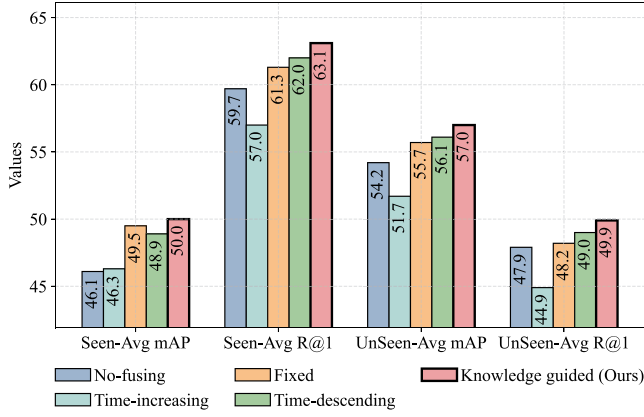


Fig. 12. Effectiveness of different model fusion strategies.

knowledge and hinder the correct knowledge transfer. Compared to the other two methods, our rectification strategy not only corrects the erroneous knowledge but also adequately retrains the original relation distribution.

Ablation on the choices model fusion strategy in $\mathcal{M}_e$: To demonstrate the effectiveness of our knowledge-guided model fusion strategy in $\mathcal{M}_e$, we compare it against several commonly used choices including (a) a fixed parameter 0.5 [39], (b) a time-increasing parameter $t - 1/t (t \geq 2)$, (c) a time-descending parameter $1/t (t \geq 2)$ [4], [73]. The results, illustrated in Fig. 12, show that our knowledge-guided model fusion strategy consistently outperforms all these alternatives on both seen and unseen datasets. This demonstrates that our active balance approach more effectively consolidates valid knowledge during model fusion.

2) Ablation Studies on LSTKC++

The effectiveness of LSTKC++ components: In the lower part of Table III, we present the ablation studies of LSTKC++ components. When $\mathcal{M}_r^*$ is added to the baseline, more significant improvements on 3 out of 4 metrics are achieved compared to $\mathcal{M}_r$, verifying the effectiveness of the long-term and short-term complementary knowledge rectification mechanism. The slight underperformance in the Seen-Avg R@1 metric compared to $\mathcal{M}_r$ is due to some fused knowledge being difficult to transfer to the new model. Next, by introducing $\mathcal{R}_e^*$ into the baseline, where only a single long-term old model is preserved, we observe a notable improvement in Seen-Avg mAP and UnSeen-Avg mAP/R@1 compared to $\mathcal{R}_e$. Since $\mathcal{M}_o^*$ does not directly affect performance when used alone, we combined it with $\mathcal{R}_e^*$, and the results demonstrate that the backward rectification of the old model fusing weight promotes the learned knowledge retention. Finally, when all three modules are used together, the model achieves **3.6%–6.4%** improvements across four metrics compared to the original LSTKC method, validating the overall effectiveness of the proposed extensions.

Ablation on the choices of knowledge transfer loss: We also conduct ablation experiments to show the influence of different knowledge transfer loss functions on our LSTKC++ model. In this paper, the KL divergence $\mathcal{L}_{KL}$ is adopted as the default knowledge transfer loss function. In this section, we replace

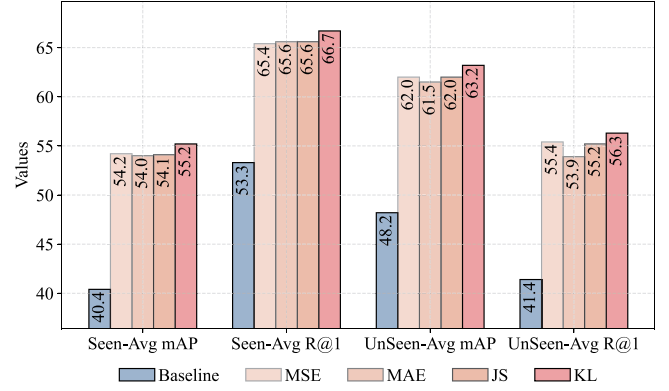


Fig. 13. Robustness of our methods to different loss functions.

$\mathcal{L}_{KL}$ with MSE [74], MAE, and Jensen-Shannon (JS) divergence [9] that are commonly used in lifelong learning. The results, as depicted in Fig. 13, demonstrate that our model can adapt well to different loss functions, yielding a significant improvement over the Baseline. Among these loss functions, KL divergence performs the best due to its specific design for knowledge transfer from the target distribution to the source distribution, aligning closely with the objective of LSTKC++.

Influence of mini-batch setting: Although existing works suggest that mini-batch settings have little impact on LReID performance [72], we discover that an appropriate mini-batch setting is a valuable factor in facilitating knowledge consolidation capacity in our LSTKC++. We have conducted extensive experiments using different configurations for identity numbers (Num-IDs) and image numbers per ID (Num-Imgs). The results in Table IV show that relatively smaller Num-IDs and larger Num-Imgs compared to the famous 32-IDs and 4-Imgs setting can improve model performance on both seen and unseen domains. Specifically, the mini-batch configurations of 8ID-8Img and 4ID-16Img yield the best performance on the seen and unseen domains, respectively.

This observation arises from our relation-based knowledge rectification and transfer design, wherein relatively larger Num-Imgs facilitate the inclusion of more local relation knowledge in the feature space. In contrast, larger Num-IDs can introduce long-distance feature relations that have minimal impact on person matching but impose additional constraints. Nevertheless, preserving inter-ID relations remains crucial for maintaining discriminative features within the feature space. Therefore, the optimal mini-batch setting of our method is a balance of Num-IDs and Num-Imgs. The default setting of our mini-batch is highlighted in Table IV.

Investigation on Different Training Orders: Following MEGE [72], we evaluate the performance of different methods on five additional dataset orders to comprehensively demonstrate the robustness of the proposed approach. The results are presented in Fig. 14(a)–(e). Our LSTKC++ consistently achieves superior performance across all training orders compared to the existing LReID works. Besides, in Fig. 14(f), we visualize the average performance across the five training orders. We observe **3.3%/3.0%** and **4.6%/4.4%** average mAP/R@1 improvements

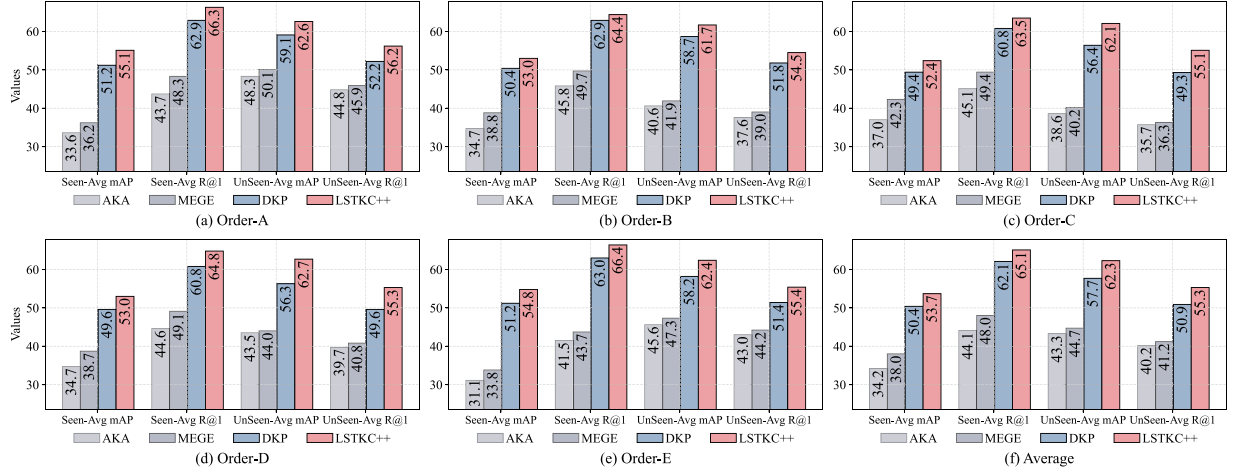


Fig. 14. Performance comparison on five more training orders. (a) Order-A: MS→SY→DU→MA→CU. (b) Order-B: DU→MA→CU→MS→SY. (c) Order-C: SY→DU→CU→MS→MA. (d) Order-D: CU→MS→DU→MA→SY. (e) Order-E: MA→MS→DU→SY→CU. (f) The average performance of the above five orders. MS, SY, DU, MA, and CU are abbreviations for MSMT17, CUHK-SYSU, DukeMTMC, Market-1501, and CUHK03, respectively.

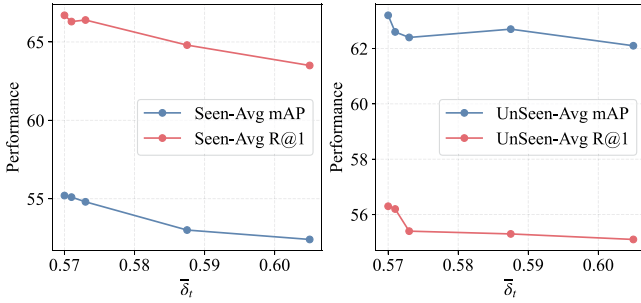


Fig. 15. Visualization of model performance under varying average δ_t ($\bar{\delta}_t$) across different domain orders.

over the state-of-the-art DKP on the seen and unseen domains, respectively. These results highlight the effectiveness of our model in handling diverse and varying domains.

Furthermore, we observe that performance fluctuations occur across different training orders for all LReID approaches. To investigate this phenomenon, we analyze the relationship between performance and the average domain shift encountered during lifelong training. Given that δ_t calculated by (15) indicates the domain gap between two adjacent datasets, we compute the average δ_t , denoted as $\bar{\delta}_t$, to quantify the overall domain shift within each training order. As illustrated in Fig. 15, model performance on both seen and unseen domains exhibits a negative correlation with $\bar{\delta}_t$. This suggests that larger domain shifts exacerbate forgetting during knowledge transfer, thereby degrading overall performance. Notably, unseen domain performance is less influenced by $\bar{\delta}_t$ compared to seen domain performance, as generalizable knowledge is less dependent on specific training data, making it more robust to variations in training order.

Evaluation of Training Cost: In Table V, we report the Batch time, Model Memory cost, Parameters (Params), and GPU Memory cost of our method compared to the latest LReID methods, all evaluated on the same device with a batch size of 128. Specifically, regarding **Batch Time**, our LSTKC is the most efficient, as matrix computation is significantly faster

than the additional modules employed in existing methods, such as patch sampling [10], graph networks [4] and instance sampling [39], used in the existing works. Our LSTK++ requires approximately 30% more batch time than LSTKC due to the introduction of an additional old model. In terms of **Params**, both LSTKC and LSTK++ possess the least number of trainable parameters, as no additional learnable components are introduced beyond the backbone and classifier. In contrast, previous LReID methods [4], [10], [39] typically incorporate additional learnable modules, increasing parameter complexity. For **Model Memory**, LSTKC exhibits the lowest memory consumption since only the backbone and classifier are stored. Existing works typically require additional storage for learnable parameters [4], [10], [39], graph nodes [4], prototypes [39]. LSTK++ doubles the memory usage of LSTKC, as the single old model is decomposed into long-term and short-term old models. Regarding **GPU Memory**, LSTKC and LSTK++ are the first and second most efficient methods, respectively. The lightweight relation matrix enables minimal GPU memory consumption, while additional networks and operations in existing approaches significantly increase computational overhead. Notably, although LSTK++ employs an extra old model, it only introduces an additional 818 MB (a 7.4% increase) in GPU memory usage compared to LSTKC. This is because the old model remains frozen, meaning its gradients are neither calculated nor stored during new model training, leading to only a marginal increase in GPU memory consumption.

In addition, to evaluate the time cost of our linear search strategy in the Old Knowledge Backward Rectification module, we conduct experiments on the training order-1, recording both the linear search time and the model training time at each training step. The results are presented in Fig. 16. Specifically, during the initial two training steps, the linear search time remains at 0 minutes since it is not performed before training step 3. Once the training step number exceeds 2, the linear search is conducted to update the long-term old model. Notably, the time required for the linear search is significantly lower than that for model training, accounting for **less than 10%** of the latter. Despite

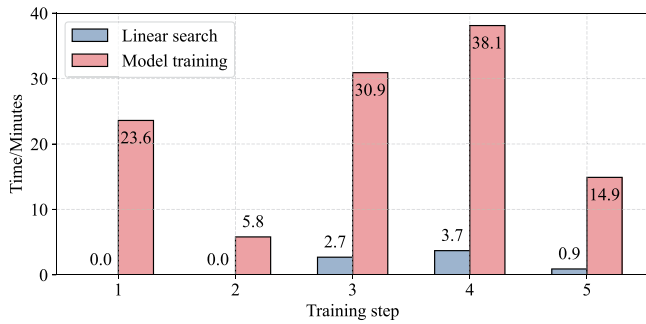


Fig. 16. Comparison of the time cost of linear search and training along the training steps. The experiments are conducted on training order-1.

the marginal increase in total learning time, our LSTKC++ achieves substantial improvements of 5.2%/4.6% in Seen-Avg mAP/R@1 and 6.2%/6.4% in Unseen-Avg mAP/R@1 under training order-1 compared to LSTKC, as shown in Table I, demonstrating the effectiveness of our approach.

VI. CONCLUSION

In this paper, to tackle the challenging non-exemplar LReID task, we propose a novel approach named Long Short-Term Knowledge Decomposition and Consolidation (LSTKC++). Our method pioneers a new paradigm in LReID. On the one hand, the erroneous knowledge from old models is rectified to mitigate its interference with the short-term learning of new knowledge. On the other hand, the new and old models are adaptively fused to achieve better acquisition and forgetting balancing. To further enhance the knowledge consolidation capacity, we propose to decompose a single old model into a long-term old model and a short-term old model. A backward rectification mechanism is designed to improve the old knowledge balancing based on the decomposition. Moreover, a complementary knowledge mining and rectification strategy is proposed to ensure the comprehensive utilization of the decomposed models. The extensive experimental results and statistics show that our LSTKC++ requires less overhead but outperforms the state-of-the-art methods by large margins.

REFERENCES

- [1] L. Zheng, L. Shen, L. Tian, S. Wang, J. Wang, and Q. Tian, "Scalable person re-identification: A benchmark," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 1116–1124.
- [2] Y. Zhao et al., "Learning to generalize unseen domains via memory-based multi-source meta-learning for person re-identification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 6277–6286.
- [3] L. Fu, Y. Du, Y. Ding, D. Wang, H. Jiang, and H. Zhang, "Domain adaptive learning with multi-granularity features for unsupervised person re-identification," *Chin. J. Electron.*, vol. 31, no. 1, pp. 116–128, 2022.
- [4] N. Pu, W. Chen, Y. Liu, E. M. Bakker, and M. S. Lew, "Lifelong person re-identification via adaptive knowledge accumulation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 7897–7906.
- [5] M. Liang, J. Zhou, W. Wei, and Y. Wu, "Balancing between forgetting and acquisition in incremental subpopulation learning," in *Proc. Eur. Conf. Comput. Vis.*, Springer, 2022, pp. 364–380.
- [6] R. Zhang, E. Haihong, M. Song, and X. Cao, "FSCIL-EACA: Few-shot class-incremental learning network based on embedding augmentation and classifier adaptation for image classification," *Chin. J. Electron.*, vol. 33, no. 1, pp. 139–152, 2024.
- [7] G. Wu and S. Gong, "Generalising without forgetting for lifelong person re-identification," in *Proc. AAAI Conf. Artif. Intell.*, 2021, pp. 2889–2897.
- [8] W. Ge et al., "Lifelong person re-identification by pseudo task knowledge preservation," in *Proc. AAAI Conf. Artif. Intell.*, 2022, pp. 688–696.
- [9] C. Yu, Y. Shi, Z. Liu, S. Gao, and J. Wang, "Lifelong person re-identification via knowledge refreshing and consolidation," in *Proc. AAAI Conf. Artif. Intell.*, 2023, pp. 3295–3303.
- [10] Z. Sun and Y. Mu, "Patch-based knowledge distillation for lifelong person re-identification," in *Proc. ACM Int. Conf. Multimedia*, 2022, pp. 696–707.
- [11] K. Xu, X. Zou, and J. Zhou, "LSTKC: Long short-term knowledge consolidation for lifelong person re-identification," in *Proc. AAAI Conf. Artif. Intell.*, 2024, pp. 16 202–16 210.
- [12] E. Ahmed, M. Jones, and T. K. Marks, "An improved deep learning architecture for person re-identification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 3908–3916.
- [13] W. Li, X. Zhu, and S. Gong, "Harmonious attention network for person re-identification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 2285–2294.
- [14] H. Luo, Y. Gu, X. Liao, S. Lai, and W. Jiang, "Bag of tricks and a strong baseline for deep person re-identification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Worksh.*, 2019, pp. 1487–1495.
- [15] Z. Ye, X. He, and Y. Peng, "Unsupervised cross-media hashing learning via knowledge graph," *Chin. J. Electron.*, vol. 31, no. 6, pp. 1081–1091, 2022.
- [16] L. Wei, S. Zhang, W. Gao, and Q. Tian, "Person transfer GAN to bridge domain gap for person re-identification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 79–88.
- [17] Z. Zhuang et al., "Rethinking the distribution gap of person re-identification with camera-based batch normalization," in *Proc. Eur. Conf. Comput. Vis.*, Springer, 2020, pp. 140–157.
- [18] S. He, H. Luo, P. Wang, F. Wang, H. Li, and W. Jiang, "TransReID: Transformer-based object re-identification," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2021, pp. 14 993–15 002.
- [19] Y.-C. Chen, X. Zhu, W.-S. Zheng, and J.-H. Lai, "Person re-identification by camera correlation aware feature augmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 2, pp. 392–408, Feb. 2018.
- [20] Z. Zhong, L. Zheng, Z. Zheng, S. Li, and Y. Yang, "Camera style adaptation for person re-identification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 5157–5166.
- [21] X. Zhu, X. Zhu, M. Li, P. Morerio, V. Murino, and S. Gong, "Intra-camera supervised person re-identification," *Int. J. Comput. Vis.*, vol. 129, pp. 1580–1595, 2021.
- [22] J. Liu, Z.-J. Zha, D. Chen, R. Hong, and M. Wang, "Adaptive transfer network for cross-domain person re-identification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 7195–7204.
- [23] J. Song, Y. Yang, Y.-Z. Song, T. Xiang, and T. M. Hospedales, "Generalizable person re-identification by domain-invariant mapping network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 719–728.
- [24] X. Jin, C. Lan, W. Zeng, Z. Chen, and L. Zhang, "Style normalization and restitution for generalizable person re-identification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 3143–3152.
- [25] Y. Zou, X. Yang, Z. Yu, B. V. Kumar, and J. Kautz, "Joint disentangling and adaptation for cross-domain person re-identification," in *Proc. Eur. Conf. Comput. Vis.*, Springer, 2020, pp. 87–104.
- [26] K. Zheng, C. Lan, W. Zeng, Z. Zhang, and Z.-J. Zha, "Exploiting sample uncertainty for domain adaptive person re-identification," in *Proc. AAAI Conf. Artif. Intell.*, 2021, pp. 3538–3546.
- [27] W. Deng, L. Zheng, Q. Ye, G. Kang, Y. Yang, and J. Jiao, "Image-image domain adaptation with preserved self-similarity and domain-dissimilarity for person re-identification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 994–1003.
- [28] L. Qi, L. Wang, J. Huo, L. Zhou, Y. Shi, and Y. Gao, "A novel unsupervised camera-aware domain adaptation framework for person re-identification," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 8080–8089.
- [29] H. Feng, M. Chen, J. Hu, D. Shen, H. Liu, and D. Cai, "Complementary pseudo labels for unsupervised domain adaptation on person re-identification," *IEEE Trans. Image Process.*, vol. 30, pp. 2898–2907, 2021.
- [30] F. Zhao, S. Liao, G.-S. Xie, J. Zhao, K. Zhang, and L. Shao, "Unsupervised domain adaptation with noise resistible mutual-training for person re-identification," in *Proc. Eur. Conf. Comput. Vis.*, Springer, 2020, pp. 526–544.
- [31] G. Chen, Y. Lu, J. Lu, and J. Zhou, "Deep credible metric learning for unsupervised domain adaptation person re-identification," in *Proc. Eur. Conf. Comput. Vis.*, Springer, 2020, pp. 643–659.

- [32] Z. Wang et al., "DualPrompt: Complementary prompting for rehearsal-free continual learning," in *Proc. Eur. Conf. Comput. Vis.*, Springer, 2022, pp. 631–648.
- [33] P. Mazumder, P. Singh, and P. Rai, "Few-shot lifelong learning," in *Proc. Conf. Assoc. Advance. Artif. Intell.*, 2021, pp. 2337–2345.
- [34] P. Singh, V. K. Verma, P. Mazumder, L. Carin, and P. Rai, "Calibrating CNNs for lifelong learning," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2020, pp. 15 579–15 590.
- [35] K. Shmelkov, C. Schmid, and K. Alahari, "Incremental learning of object detectors without catastrophic forgetting," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 3420–3429.
- [36] Y. Wang, Z. Huang, and X. Hong, "S-prompts learning with pre-trained transformers: An occam's razor for domain incremental learning," 2022, *arXiv:2207.12819*.
- [37] S.-A. Rebuffi, A. Kolesnikov, G. Sperl, and C. H. Lampert, "iCaRL: Incremental classifier and representation learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 5533–5542.
- [38] B. Zhao, X. Xiao, G. Gan, B. Zhang, and S.-T. Xia, "Maintaining discrimination and fairness in class incremental learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 13 205–13 214.
- [39] K. Xu, X. Zou, Y. Peng, and J. Zhou, "Distribution-aware knowledge prototyping for non-exemplar lifelong person re-identification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2024, pp. 16 604–16 613.
- [40] Z. Huang et al., "Lifelong unsupervised domain adaptive person re-identification with coordinated anti-forgetting and adaptation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 14 288–14 297.
- [41] J. Huang et al., "Learning consistent region features for lifelong person re-identification," *Pattern Recognit.*, vol. 144, 2023, Art. no. 109837.
- [42] N. Pu, Y. Liu, W. Chen, E. M. Bakker, and M. S. Lew, "Meta reconciliation normalization for lifelong person re-identification," in *Proc. ACM Int. Conf. Multimedia*, 2022, pp. 541–549.
- [43] Z. Yang, D. Wu, W. Zhang, B. Li, and W. Wang, "Handling label uncertainty for camera incremental person re-identification," in *Proc. ACM Int. Conf. Multimedia*, 2023, pp. 6253–6263.
- [44] K. Xu, H. Zhang, Y. Li, Y. Peng, and J. Zhou, "Mitigate catastrophic remembering via continual knowledge purification for noisy lifelong person re-identification," in *Proc. ACM Int. Conf. Multimedia*, 2024, pp. 5790–5799.
- [45] Q. Li, K. Xu, Y. Peng, and J. Zhou, "Exemplar-free lifelong person re-identification via prompt-guided adaptive knowledge consolidation," *Int. J. Comput. Vis.*, vol. 132, no. 11, pp. 4850–4865, 2024.
- [46] Z. Hu, Y. Li, J. Lyu, D. Gao, and N. Vasconcelos, "Dense network expansion for class incremental learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 11 858–11 867.
- [47] H. Wen et al., "Class incremental learning with multi-teacher distillation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2024, pp. 28 443–28 452.
- [48] D.-W. Zhou, H.-L. Sun, H.-J. Ye, and D.-C. Zhan, "Expandable sub-space ensemble for pre-trained model-based class-incremental learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2024, pp. 23 554–23 564.
- [49] J. He, "Gradient reweighting: Towards imbalanced class-incremental learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2024, pp. 16 668–16 677.
- [50] Z. Li and D. Hoiem, "Learning without forgetting," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 12, pp. 2935–2947, Dec. 2018.
- [51] Z. Luo, Y. Liu, B. Schiele, and Q. Sun, "Class-incremental exemplar compression for class-incremental learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 11 371–11 380.
- [52] P. Singh, P. Mazumder, P. Rai, and V. P. Namboodiri, "Rectification-based knowledge retention for continual learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 15 282–15 291.
- [53] J. Liu, L. Song, and Y. Qin, "Prototype rectification for few-shot learning," in *Proc. Eur. Conf. Comput. Vis.*, Springer, 2020, pp. 741–756.
- [54] Z. Cui, Y. Peng, X. Wang, M. Zhu, and J. Zhou, "Continual vision-language retrieval via dynamic knowledge rectification," in *Proc. AAAI Conf. Artif. Intell.*, 2024, pp. 11 704–11 712.
- [55] H. Liu, J. Feng, M. Qi, J. Jiang, and S. Yan, "End-to-end comparative attention networks for person re-identification," *IEEE Trans. Image Process.*, vol. 26, no. 7, pp. 3492–3506, Jul. 2017.
- [56] E. Ristani, F. Solera, R. Zou, R. Cucchiara, and C. Tomasi, "Performance measures and a data set for multi-target, multi-camera tracking," in *Proc. Eur. Conf. Comput. Vis.*, Springer, 2016, pp. 17–35.
- [57] T. Xiao, S. Li, B. Wang, L. Lin, and X. Wang, "End-to-end deep learning for person search," 2016, *arXiv:1604.01850*.
- [58] W. Li, R. Zhao, T. Xiao, and X. Wang, "DeepReID: Deep filter pairing neural network for person re-identification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 152–159.
- [59] W. Li, R. Zhao, and X. Wang, "Human reidentification with transferred metric learning," in *Proc. Asian. Conf. Comput. Vis.*, Springer, 2012, pp. 31–44.
- [60] W. Li and X. Wang, "Locally aligned feature transforms across views," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2013, pp. 3594–3601.
- [61] D. Gray and H. Tao, "Viewpoint invariant pedestrian recognition with an ensemble of localized features," in *Proc. Eur. Conf. Comput. Vis.*, Springer, 2008, pp. 262–275.
- [62] M. Hirzer, C. Belezni, P. M. Roth, and H. Bischof, "Person re-identification by descriptive and discriminative classification," in *Image Analysis*. Berlin, Germany: Springer, 2011, pp. 91–102.
- [63] H. O. S. D. Branch, "Imagery library for intelligent detection systems (i-lids)," in *Proc. 2006 IET Conf. Crime Secur.*, 2006, pp. 445–448.
- [64] C. C. Loy, T. Xiang, and S. Gong, "Time-delayed correlation analysis for multi-camera activity understanding," *Int. J. Comput. Vis.*, vol. 90, no. 1, pp. 106–129, 2010.
- [65] H. Zhao et al., "Spindle net: Person re-identification with human body region guided feature decomposition and fusion," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 907–915.
- [66] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.
- [67] F. Tung and G. Mori, "Similarity-preserving knowledge distillation," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 1365–1374.
- [68] N. Asadi, M. Davari, S. Mudur, R. Aljundi, and E. Belilovsky, "Prototype-sample relation distillation: Towards replay-free continual learning," in *Proc. Int. Conf. Mach. Learn.*, PMLR, 2023, pp. 1093–1106.
- [69] W. Shi and M. Ye, "Prototype reminiscence and augmented asymmetric knowledge aggregation for non-exemplar class-incremental learning," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2023, pp. 1772–1781.
- [70] Q. Li, Y. Peng, and J. Zhou, "FCS: Feature calibration and separation for non-exemplar class incremental learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2024, pp. 28 495–28 504.
- [71] B. Zhao, S. Tang, D. Chen, H. Bilen, and R. Zhao, "Continual representation learning for biometric identification," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, 2021, pp. 1197–1207.
- [72] N. Pu, Z. Zhong, N. Sebe, and M. S. Lew, "A memorizing and generalizing framework for lifelong person re-identification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 11, pp. 13 567–13 585, Nov. 2023.
- [73] G. Lin, H. Chu, and H. Lai, "Towards better plasticity-stability trade-off in incremental learning: A simple linear connector," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 89–98.
- [74] W. Park, D. Kim, Y. Lu, and M. Cho, "Relational knowledge distillation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 3967–3976.



Kunlun Xu received the BE and MS degrees from the Huazhong University of Science and Technology, in 2020 and 2023, respectively. Currently, he is working toward the PhD degree with the Wangxuan Institute of Computer Technology, Peking University. His current research interests include computer vision, lifelong learning, and person re-identification. He has authored 12 papers in international journals and conferences, including IJCV, ICML, CVPR, AAAI, ACM MM, and so on. He serves as a regular reviewer member for a number of journals and conferences, e.g., *IEEE Transactions on Image Processing*, *IEEE Transactions on Multimedia*, *Pattern Recognition Letters*, *ICML*, *NeurIPS*, *CVPR*, *ICCV*, *ACM MM*, and so on.



Zichen Liu received the BE degree from the Faculty of Electronic and Information Engineering, Xi'an Jiaotong University, Xi'an, China, in 2022. He is currently working toward the MS degree with the Wangxuan Institute of Computer Technology, Peking University. He is interested in visual prompting and test-time adaptation.



Xu Zou (Member, IEEE) received the PhD degree from the Huazhong University of Science and Technology, Wuhan, China, in 2020. He is currently a lecturer with the School of Artificial Intelligence and Automation, Huazhong University of Science and Technology. His research interests include image processing, computer vision, and intelligent system.



Yuxin Peng (Senior Member, IEEE) received the PhD degree in computer application technology from Peking University, Beijing, China, in 2003. He is the Boya distinguished professor with the Wangxuan Institute of Computer Technology, Peking University. He was a recipient of the National Science Fund for Distinguished Young Scholars of China. His research interests mainly include cross-media analysis and reasoning, image and video recognition and understanding, and computer vision. He has authored more than 200 papers, including more than 100 papers in top-tier

journals and conference proceedings. He has submitted 51 patent applications and has been granted 39 of them. He led his team to win the First Place in the video semantic search evaluation of TRECVID ten times in recent years. He won the First Prize of the Beijing Science and Technology Award in 2016 (ranking first) and the First Prize of the Scientific and Technological Progress Award of the Chinese Institute of Electronics in 2020 (ranking first). He was a recipient of the Best Paper award at MMM 2019 and NCIG 2018, and serves as the associate editor for *IEEE Transactions on Multimedia*, *IEEE Transactions on Circuits and Systems for Video Technology*, etc.



Jiahuan Zhou received the BE from Tsinghua University, in 2013, and the PhD degree from the Department of Electrical Engineering & Computer Science, Northwestern University, in 2018. Currently, he is a Tenure-Track assistant professor with the Wangxuan Institute of Computer Technology, Peking University. His research interests include computer vision, deep learning, and machine learning. He has authored more than 60 papers in international journals and conferences including *Nature Communications*, *Nature Synthesis*, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *International Journal of Computer Vision*, *IEEE Transactions on Image Processing*, *IEEE Transactions on Information Forensics and Security*, *CVPR*, *ICCV*, *ICML*, *ECCV*, *ICLR*, *AAAI*, *ACM MM*, and so on. He serves as an area chair for *CVPR*, *ICML*, *NeurIPS*, *ICME*, and *ICPR*, an associate editor for *Springer Journal of Machine Vision and Applications* (MVA), a regular reviewer member for several journals and conferences, e.g., *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *IJCV*, *TIP*, *CVPR*, *ICCV*, *ECCV*, *NeurIPS*, *ICML*, *ICLR*, *AAAI*, and so on.