Augmented Geometric Distillation for Data-Free Incremental Person ReID

Yichen Lu Mei Wang Weihong Deng*
Beijing University of Posts and Telecommunications
{yichen.lu, wangmei1, whdeng}@bupt.edu.cn

Abstract

Incremental learning (IL) remains an open issue for Person Re-identification (ReID), where a ReID system is expected to preserve preceding knowledge while learning incrementally. However, due to the strict privacy licenses and the open-set retrieval setting, it is intractable to adapt existing class IL methods to ReID. In this work, we propose an Augmented Geometric Distillation (AGD) framework to tackle these issues. First, a general data-free incremental framework with dreaming memory is constructed to avoid privacy disclosure. On this basis, we reveal a "noisy distillation" problem stemming from the noise in dreaming memory, and further propose to augment distillation in a pairwise and cross-wise pattern over different views of memory to mitigate it. Second, for the open-set retrieval property, we propose to maintain feature space structure during evolving via a novel geometric way and preserve relationships between exemplars when representations drift. Extensive experiments demonstrate the superiority of our AGD to baseline with a margin of 6.0% mAP / 7.9% R@1 and it could be generalized to class IL. Code is available here†.

1. Introduction

Person re-identification (ReID) aims at identifying all images of the same person as the query from a gallery set of large scale. Training on a certain dataset empirically empowers a ReID system to expert in the corresponding domain. However, it inhibits the ReID system from adapting to the ever-changing environment, especially when dealing with the streamed data or a sequence of ReID tasks from *incremental* domains. We expect the system can widen its generalization in incremental domain and retain its capability in base domain simultaneously, which is, briefly, to accumulate new knowledge while avoiding Catastrophic Forgetting [12, 29]. To overcome such similar limitation, Class Incremental Learning (CIL) [5, 10, 18, 24, 33, 44] is proposed

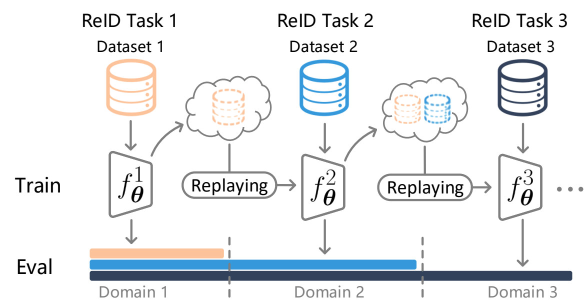

Figure 1. Illustration of data-free IL-ReID framework. The model keeps evolving when training on a sequence of ReID tasks. Evaluation is adopted in all seen domains. Replaying is in a data-free setting [37, 47] due to privacy issues in ReID, where no preceding real data is stored, instead, dreaming memory drives relaying.

in classification task and efforts have been devoted to figuring out how to learn incrementally.

Despite the great success in CIL, it still faces challenges when directly adopted to a ReID system due to the strict *privacy* issues and the *open-set retrieval* setting. First, in CIL, reminding the networks of previous knowledge via replaying pre-stored exemplars is well-recognized [5, 19, 33] to alleviate catastrophic forgetting. However, replaying memory of real data faces risks of violating privacy licenses in ReID. Second, on one side, ReID is substantially an open-set retrieval task, which puts more attention on constructing a robust feature space when compared with the close-set classification, since not only representations but also their neighborhoods play key roles in retrieval ranking. On the other side, feeding new knowledge sequentially will inevitably cause semantic drift [49] and distort the preceding feature space, resulting in forgetting. Hence, there exists a critical yet ignored contradiction between stabilizing the feature space for preceding domains and adapting feature space for the incremental domain.

Considering the limitations aforementioned, we conduct further research on Incremental ReID (IL-ReID) [32] and propose a novel Augmented Geometric Distillation (AGD) framework which consists of Augmented Distillation (AD) and Geometric Distillation (GD). First, to tackle the privacy issue, we first construct a general data-free incre-

*Corresponding author
†https://github.com/eddielyc/Augmented-Geometric-Distillation

mental framework for IL-ReID (overview in Fig. 1), in which *dreaming memory*, generated by DeepInversion [47], drives replaying procedure without access to preceding real data. Unfortunately, due to the poor quality, directly replaying these dreaming exemplars will induce a phenomenon termed "noisy distillation", during which, noisy knowledge will be transferred into the evolving model and aggravate forgetting. To alleviate this problem, we further propose to augment distillation itself. Enlightened by contrastive learning, we produce different views of memory and distill in a pair-wise and cross-wise pattern to strengthen the robustness and reduce the perturbation.

Second, to handle the contradiction caused by open-set retrieval property, we propose the geometric distillation (GD) tailored-made for retrieval task that our intuition is to maintain the structure of the preceding feature space while drifting instead of to stabilize the whole space and to penalize drift. The structure of preceding space is formulated with exemplars in dreaming memory. To prevent exemplars from drifting in their own manners arbitrarily and "roiling" the space structure, we encourages exemplars to drift in a consistent manner, so that the structure could be maintained via similarity criterion in a novel geometric way. This allows to adapt the feature space for new knowledge while preserving rich preceding information for retrieval, offering a compromise between learning and memorizing.

To conclude, our contributions could be summarized as:

- **i)** We construct a data-free incremental framework for ReID with dreaming memory. It serves without privacy issues;
- **ii)** We propose Augmented Distillation (AD), where distillation is conducted in a pair-wise and cross-wise pattern to address the "noisy distillation" phenomenon in dreaming memory;
- **iii)** We propose Geometric Distillation (GD) to adapt new and preceding knowledge for retrieval tasks via maintaining space structure geometrically when drifting;
- **iv)** We adapt mainstream solutions in CIL to ReID. Extensive experiments indicate that our AGD is superior to baseline with a margin of 6.0% mAP / 7.9% R@1 and it is promising to be generalized to CIL.

2. Related Work

2.1. Incremental Learning

Incremental learning [41] studies the problem of accumulating knowledge sequentially without catastrophic forgetting [12, 34]. To achieve this, methods based on parameters regularization [2, 23] attempted to penalize updating parameters for preceding tasks. Parameter-isolation based methods [1,22,28] dedicated extra parameters for new tasks. The recent mainstreams are on insights of replaying memory and distilling knowledge. LwF [24] first introduced distillation into IL. Dhar *et al.* [9] further proposed to constrain attention. iCaRL [33] and its improved variants [5, 18] introduced replay mechanism, where a memorizer is maintained to store limited samples for replaying. Following up on this, Wu *et al.* [44] and Hou *et al.* [19] corrected the bias in classifier. PODNet [10] distilled the pooled intermediate feature maps and GeoDL [36] constrained the geodesic flow in lower dimensions. TOPIC [40] and TPCIL [39] put their emphasis on topology of exemplars. Despite the remarkable insights, a compact *memorizer* is indispensable to all these replaying-based methods. As data-free frameworks, ARM [21] and ABD [37] replayed generated memory instead, but "noisy distillation" is ignored. SDC [49] measured the semantic drift without memory, but it was oriented to classification not retrieval.

2.2. Data-free Knowledge Transfer

As the seminal work by Hinton *et al.* [17], a basic solution was proposed to compress knowledge to student networks, on which to base, a line of works [25, 45, 46, 50] has reported more effective solutions. However, most methods above are data-driven. To address this flaw, some works managed to generate images. Lopes *et al.* [26] synthesized images via meta-data of networks. Bhardwaj *et al.* [3] synthesized samples via pre-recorded the centroids of classes. Some works [4, 14, 47] discovered constraining the generated images to match BatchNorm [20] statistics in teacher networks could close the gap between real image distribution. Similarly, Yoo *et al.* [48] and Chen *et al.* [6] trained a decoder to output class-conditional images. Moreover, some papers [8, 11, 30] transferred knowledge in an adversarial strategy. Despite notable progress above, how to retain knowledge in the pre-trained base model and incrementally learn from new tasks remains under-explored.

3. Background and Data-free Framework

In this section, we define the IL-ReID (Sec. 3.1) and clarify a data-free incremental framework for ReID (Sec. 3.2).

3.1. Problem Definition

In IL-ReID, to provide basic knowledge, T_1 leads the task sequence. Following the setting in LUCIR [19], the first task T_1 contains samples with a wide variety to achieve a strong base model $f_{\boldsymbol{\theta}}^1$. After that, similar to the *task*-incremental setting [31] in CIL, data from a sequence of ReID tasks T_2, T_3, T_4 ... will be continually presented for incremental learning. In incremental training stage of T_n, the base model $f_{\boldsymbol{\theta}}^{n-1}$ evolves into $f_{\boldsymbol{\theta}}^n$. During this, we can only access to base model $f_{\boldsymbol{\theta}}^{n-1}$ and dataset of T_n. Especially, no real data of base tasks $T_{1:n-1} = \{T_i\}_{i=1}^{n-1}$ is available. The dataset of T_n is denoted as $\mathcal{D}_{T_n} = \{(\boldsymbol{x}_i^n, y_i^n)\}_{i=1}^{N_n}$, where $(\boldsymbol{x}_i^n, y_i^n)$ is the *i*-th image and its ID. N_n is the number of images in $\mathcal{D}_{T_n}$.

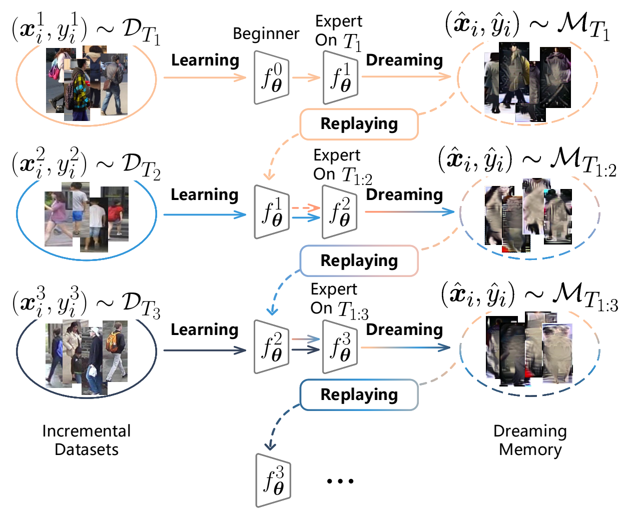

Figure 2. Pipeline of our data-free framework. In n-th incremental step, **Dreaming:** Generate dreaming memory $\mathcal{M}_{T_{1:n-1}}$, **Learning:** Learn new knowledge from $\mathcal{D}_{T_N}$, **Replaying:** Replay memory $\mathcal{M}_{T_{1:n-1}}$, generally with distillation.

Objectives. When the base model $f_{\boldsymbol{\theta}}^{n-1}$ evolves into $f_{\boldsymbol{\theta}}^{n}$, we expect that **i):** accumulated knowledge in base tasks $T_{1:n-1}$ should be retained as much as possible; **ii):** based on pre-trained $f_{\boldsymbol{\theta}}^{n-1}$, $f_{\boldsymbol{\theta}}^{n}$ should learn better representations on incremental task T_n. In short, after training on T_n, $f_{\boldsymbol{\theta}}^{n}$ should perform well on all seen domains in $T_{1:n}$.

3.2. Data-free Incremental Framework

To circumvent privacy issues, in contrast to storing the real data as memory, the fixed base model $f_{\boldsymbol{\theta}}^{n-1}$ "dreams" of the images over preceding image distributions.

Dreaming Memory and Replay. Dreaming memory is built via DeepInversion [47]. To synthesize images, the inputs $\hat{\boldsymbol{x}}$ are optimized to encourage the fixed base model $f_{\boldsymbol{\theta}}^{n-1}$ to output the corresponding labels $\hat{y}$. During optimization, inputs are regularized to match the BatchNorm in $f_{\boldsymbol{\theta}}^{n-1}$ to approximate over preceding distributions. After generation, the dreaming memory $\mathcal{M}_{T_{1:n-1}} = \{(\hat{\boldsymbol{x}}_i, \hat{y}_i)\}$ will be built by base model $f_{\boldsymbol{\theta}}^{n-1}$ to preserve preceding knowledge via replaying and distillation.

Representation Learning. Following basic representation training in most ReID researches, given the incremental data $\boldsymbol{x}$ from $\mathcal{D}_{T_n}$ and dreaming data $\hat{\boldsymbol{x}}$ from $\mathcal{M}_{T_{1:n-1}}$, the model should classify the inputs correctly with cross entropy loss $\mathcal{L}_{ce}$ and separate class boundaries with triplet loss $\mathcal{L}_{tri}$. Together, we formulate representation loss as follow:

$$\mathcal{L}_{rep}([\boldsymbol{x} \| \hat{\boldsymbol{x}}]) = \mathcal{L}_{ce}([\boldsymbol{x} \| \hat{\boldsymbol{x}}]) + \mathcal{L}_{tri}([\boldsymbol{x} \| \hat{\boldsymbol{x}}]), \quad (1)$$

where $[\cdot \| \cdot]$ denotes data concatenation in batch axis.

Framework Objective. The pipeline of our framework is illustrated in Fig. 2. We combine representation learning and replaying to formulate the basic framework objective:

$$\mathcal{L}_{base}(\boldsymbol{x}, \hat{\boldsymbol{x}}) = \mathcal{L}_{rep}([\boldsymbol{x} \| \hat{\boldsymbol{x}}]) + \lambda \mathcal{L}_{\kappa}(\hat{\boldsymbol{x}}), \quad (2)$$

where $\mathcal{L}_{\kappa}(\cdot)$ is the knowledge distillation term in replaying.

4. Proposed Method

Based on the design above, we propose a novel Augmented Geometric Distillation (AGD) framework to **i):** tackle the problem of "noisy distillation" in dreaming memory; **ii):** and learn to adapt knowledge flexibly yet retentively via a geometric way for IL-ReID. In Sec. 4.1, we elaborate why noise is made, visualize how it impacts the distillation and how to alleviate it via our augmented distillation. In Sec. 4.2, we pave a brand-new path to retain knowledge when representations drift via geometrically maintaining structure in euclidean feature space.

4.1. Augmented Distillation

As discussed above, in order to circumvent privacy disclosure, we adopt dreaming data to acts as the memory. We expect data generated by DeepInversion serves as effectively as real data. However, a drawback of it is that mimicking the real image distribution imperfectly causes domain gap. For instance, an evident gap in visual level between dreaming exemplars and raw images exists due to the poor quality. And such domain gap weakens the robustness to data augmentation (*e.g.* crop, flip and REA [52]) as visualized in Fig. 4 (Left). The unexpected perturbation widens divergence of dreaming exemplars in feature space, which introduces overfitting of noise in typical pair-wise knowledge distillation as Fig. 4 (Right) and hence, aggravates forgetting. Even worse, such perturbation could bring more adverse impacts to our geometric distilling (detailed in Sec. 4.2) due to unstable relationships between dreaming exemplars.

Under such circumstance, the guidance from teacher is noisy, but data augmentation is necessary to promote sample diversity. For the best of both worlds, we propose to augment distillation itself. Specifically, to mitigate perturbation in each iteration, we first follow contrastive learning [7, 13, 15] to build two views of data $\hat{\boldsymbol{x}}'$ and $\hat{\boldsymbol{x}}''$ with independent data augmentation. These views are from the same sample $\hat{\boldsymbol{x}}$ and should have robust features extracted by the teacher $f_{\boldsymbol{\theta}}^{n-1}$. However, due to the issue above, teacher outputs features of two views $f_{\boldsymbol{\theta}}^{n-1}(\hat{\boldsymbol{x}}')$ and $f_{\boldsymbol{\theta}}^{n-1}(\hat{\boldsymbol{x}}'')$ with divergence. To get more stable distilling effects, we average the gradients from pair-wise distillation of two views. Besides, $\hat{\boldsymbol{x}}'$ and $\hat{\boldsymbol{x}}''$ are sampled from the same distribution and form a congruent pair $(\hat{\boldsymbol{x}}', \hat{\boldsymbol{x}}'')$. For better consistence between views, we consider distilling across views symmetrically as illustrated in Fig. 3 (Left). The crisscross mechanism provides guidance from at least four views of the same

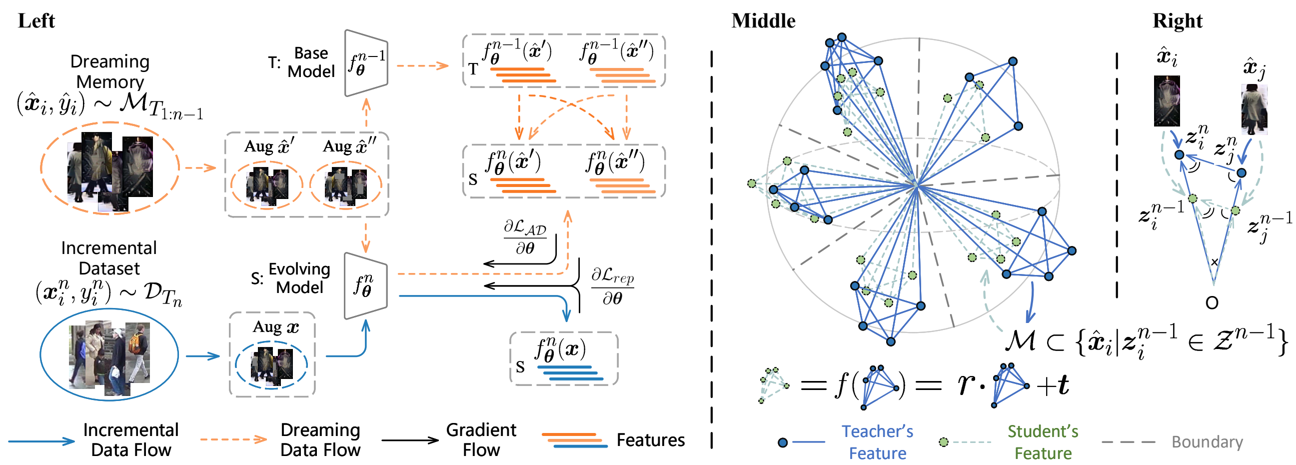

Figure 3. **Left:** Pipeline of our **Augmented Geometric Distillation** (AGD). Basic representation learning is conducted by $\mathcal{D}_{T_N}$ and $\mathcal{M}_{T_{1:N-1}}$. When preserving the knowledge, distillation process is augmented to filter out noise in dreaming data. **Middle:** Illustration of **Geometric Distillation** in Euclidean space that polyhedrons of classes built by feature points are encouraged to keep their similarity when evolving with scale and translation transforms to approximate the similarity of their subspaces. The process is driven by dreaming memory $\mathcal{M}$, which is a set of images generated by features in $\mathcal{Z}^{n-1}$. **Right:** Geometry interpretation of fundamental AAA similarity criterion.

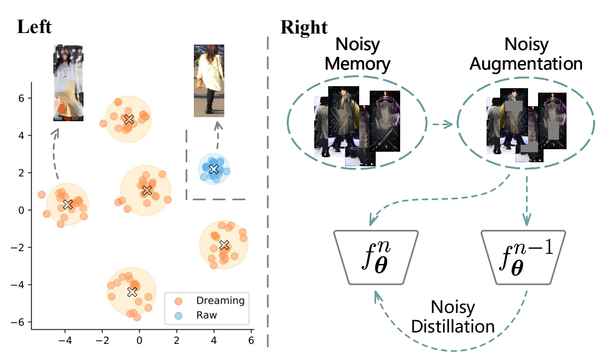

Figure 4. Visualization of noisy distillation. **Left**: In this figure, five dreaming exemplars of MSMT17 (Sec. 5.1) are augmented 20 times and features are visualized with t-SNE [42]. For comparison, a raw image of MSMT17 is processed with the same operations and much smaller divergence is observed. Points: features, crosses: cluster centroids, circle radius: divergence. **Right**: Chain reaction of noisy distillation in the typical pair-wise way.

observation to highlight the shared effective information in views and reduce the noisy part. The objective is written as:

$$\begin{aligned}&\mathcal{L}_{\mathcal{AD}}(\hat{x}', \hat{x}''; \mathcal{L}_\kappa)\\ &= \frac{1}{2}\Big[\alpha\mathcal{L}_\kappa(\hat{x}', \hat{x}') + (1-\alpha)\mathcal{L}_\kappa(\hat{x}', \hat{x}'')\\ &\quad + \alpha\mathcal{L}_\kappa(\hat{x}'', \hat{x}'') + (1-\alpha)\mathcal{L}_\kappa(\hat{x}'', \hat{x}')\Big],\end{aligned} \tag{3}$$

where $\mathcal{L}_\kappa(\hat{x}', \hat{x}'')$ calculates the distillation loss term between teacher output $f_\theta^{n-1}(\hat{x}')$ and student output $f_\theta^n(\hat{x}'')$. Other three terms are formulated similarly. α is the weight to balance pair-wise and cross-wise terms. Typically, $\mathcal{L}_\kappa(\cdot)$ could be in a form of KL divergence as iCaRL [33]:

$$\mathcal{L}_{kl}(\hat{x}) = \mathrm{KL}\big[p(y|\hat{x}, f_\theta^{n-1}), p(y|\hat{x}, f_\theta^n)\big], \tag{4}$$

or in a form of *cos* as LUCIR [19] for richer information in features:

$$\mathcal{L}_{cos}(z^{n-1}, z^n) = 1 - \langle z^{n-1}, z^n \rangle, \tag{5}$$

where $\langle \cdot, \cdot \rangle$ denotes $\cos(\cdot, \cdot)$ operation and z stands for features, *i.e.*, $z^{n-1} = f_\theta^{n-1}(\hat{x})$, $z^n = f_\theta^n(\hat{x})$.

4.2. Geometric Distillation

During incremental learning, feature space drift is inevitable and the arbitrary drift could roil the space structure (Fig. 5). Despite the success of penalizing the drift (*e.g.* Equ. 5), there exists a contradiction between preserving preceding knowledge, which detests the drift, and learning new knowledge, which leads to necessary drift. This issue bothers ReID particularly due to open-set retrieval property. To reach a compromise, we propose a brand-new solution, where space drift is not penalized explicitly. Our intuition is to maintain geometry structure of subspace of each class when drifting and keep the most discriminative representations for ranking in retrieval task. Then our method has the flexibility to fit new data and meanwhile preserves rich information in relationships with a geometric approach.

Definition 1 *Given a Euclidean space $\mathcal{Z}$, if a bijection $g(x) = rAx + t$* maps any two points x_1 and x_2 in $\mathcal{Z}$ into a Euclidean space $\mathcal{Z}'$ and $d(g(x_1), g(x_2)) = r \cdot d(x_1, x_2)$, where $d(\cdot, \cdot)$ is the Euclidean distance, we call $\mathcal{Z}'$ a similarity space to $\mathcal{Z}$ and r is the scale coefficient.*

* A is an orthogonal matrix and t is a translation vector.

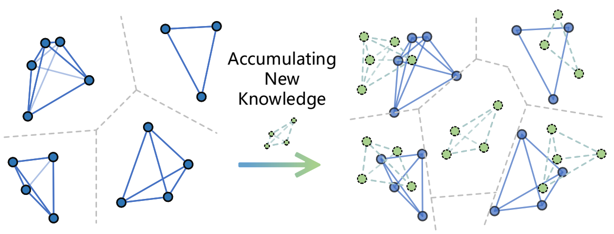

Figure 5. Visualization of arbitrary drift. When accumulating new knowledge, new features are embedded into the feature space. To adapt new knowledge, features in preceding feature spaces drift in their own manners and damage their space structures.

Geometrically, the similarity of two spaces is defined as Def. 1. Correspondingly, we expect the feature subspaces of preceding classes to maintain their structure when evolving via keeping their similarity when drifting, *i.e.*, $f_{\boldsymbol{\theta}}^{n}(\hat{\boldsymbol{x}}) = g\big(f_{\boldsymbol{\theta}}^{n-1}(\hat{\boldsymbol{x}})\big) = r\boldsymbol{A}f_{\boldsymbol{\theta}}^{n-1}(\hat{\boldsymbol{x}}) + \boldsymbol{t}$, where $\hat{\boldsymbol{x}}$ is a sample of a certain class and $f_{\boldsymbol{\theta}}^{n}(\hat{\boldsymbol{x}}) \in \mathcal{Z}^{n}$, $f_{\boldsymbol{\theta}}^{n-1}(\hat{\boldsymbol{x}}) \in \mathcal{Z}^{n-1}$. In practice, it is intractable to constrain all points in feature space. However, with the dreaming dataset $\mathcal{M}$, we can sample feature points in $\mathcal{Z}^{n-1}$. These points form polyhedrons of corresponding classes in $\mathcal{Z}^{n-1}$ and we approximate space similarity via preserving geometric structure of polyhedrons, as shown in Fig. 3 (Middle).

Starting from the basic form $f_{\boldsymbol{\theta}}^{n}(\hat{\boldsymbol{x}}) = rf_{\boldsymbol{\theta}}^{n-1}(\hat{\boldsymbol{x}})$, where $f_{\boldsymbol{\theta}}^{n}(\hat{\boldsymbol{x}})$ is the scaling of $f_{\boldsymbol{\theta}}^{n-1}(\hat{\boldsymbol{x}})$, to achieve this, we model the loss as:

$$\mathcal{L}_{\mathcal{G}}^{r}(\mathcal{Z}^{n-1},\mathcal{Z}^{n}) = \mathop{\mathbb{E}}_{(\hat{\boldsymbol{x}}_i,\hat{\boldsymbol{x}}_j)\in\mathbb{P}}\big[\mathcal{L}_{cos}(\boldsymbol{z}_i^{n-1},\boldsymbol{z}_i^{n}) + \mathcal{L}_{cos}(\boldsymbol{z}_j^{n-1},\boldsymbol{z}_j^{n}) + \mathcal{L}_{cos}(\boldsymbol{z}_i^{n-1}-\boldsymbol{z}_j^{n-1},\boldsymbol{z}_i^{n}-\boldsymbol{z}_j^{n})\big], \tag{6}$$

where $\mathbb{P}$ denotes the set of positive exemplar pairs and $\mathcal{Z}^{n} = \{\boldsymbol{z}_i^{n}|\boldsymbol{z}_i^{n} = f_{\boldsymbol{\theta}}^{n}(\hat{\boldsymbol{x}}_i)\}, \mathcal{Z}^{n-1} = \{\boldsymbol{z}_i^{n-1}|\boldsymbol{z}_i^{n-1} = f_{\boldsymbol{\theta}}^{n-1}(\hat{\boldsymbol{x}}_i)\}$. In this constrain, $\mathcal{L}_{cos}(\cdot,\cdot)$ encourages two vectors to be parallel with orientations. And three paralleled sides in Equ. 6 fullfil the AAA criterion of triangle similarity in its plane as demonstrated in Fig. 3 (Right). After enumerating all positive pairs and convergence, triangles are chained with shared sides and polyhedrons in $\mathcal{Z}^{n-1}$ gradually scale into the polyhedrons in $\mathcal{Z}^{n}$. And since the scale coefficient r is not defined in Equ. 6 explicitly, r is learnt adaptively and independently in each subspace.

Based on discussion above, we now consider the translation vector $\boldsymbol{t}$, which can be viewed as the drift of feature distribution. To allow the drift and maintain the geometric structure simultaneously, given the bijection $f_{\boldsymbol{\theta}}^{n}(\hat{\boldsymbol{x}}) = rf_{\boldsymbol{\theta}}^{n-1}(\hat{\boldsymbol{x}}) + \boldsymbol{t}$ and two samples $\hat{\boldsymbol{x}}_i, \hat{\boldsymbol{x}}_j$,

$$\begin{aligned}\Delta\boldsymbol{z}_{ij}^{n} &= \boldsymbol{z}_i^{n} - \boldsymbol{z}_j^{n} = f_{\boldsymbol{\theta}}^{n}(\hat{\boldsymbol{x}}_i) - f_{\boldsymbol{\theta}}^{n}(\hat{\boldsymbol{x}}_j) \\ &= rf_{\boldsymbol{\theta}}^{n-1}(\hat{\boldsymbol{x}}_i) - rf_{\boldsymbol{\theta}}^{n-1}(\hat{\boldsymbol{x}}_j) = r\Delta\boldsymbol{z}_{ij}^{n-1},\end{aligned} \tag{7}$$

where $\Delta\boldsymbol{z}_{ij}^{n}$ is the scale of $\Delta\boldsymbol{z}_{ij}^{n-1}$, which is a similar problem to constraining $f_{\boldsymbol{\theta}}^{n}(\hat{\boldsymbol{x}}) = rf_{\boldsymbol{\theta}}^{n-1}(\hat{\boldsymbol{x}})$. Based on Equ. 6, we formulate the constrain with feature drift as:

$$\mathcal{L}_{\mathcal{G}}^{r\boldsymbol{t}}(\mathcal{Z}^{n-1},\mathcal{Z}^{n}) = \mathcal{L}_{\mathcal{G}}^{r}(\Delta\mathcal{Z}^{n-1},\Delta\mathcal{Z}^{n}), \tag{8}$$

where $\Delta\mathcal{Z}^{n} = \{\Delta\boldsymbol{z}_{ij}^{n}|i \neq j, (\hat{\boldsymbol{x}}_i,\hat{\boldsymbol{x}}_j) \in \mathbb{P}\}$ and $\Delta\mathcal{Z}^{n-1} = \{\Delta\boldsymbol{z}_{ij}^{n-1}|i \neq j, (\hat{\boldsymbol{x}}_i,\hat{\boldsymbol{x}}_j) \in \mathbb{P}\}$. Unlike $\mathcal{L}_{cos}$ only constraining orientations of individual features, in our $\mathcal{L}_{\mathcal{G}}^{r\boldsymbol{t}}$, features scale and drift in a consistent manner, which is of great importance to maintain the relationship between intra-exemplars. Meanwhile, the scale coefficient r and translation vector $\boldsymbol{t}$ endow impressive *plasticity* when compared with much more critical criterion as $\mathcal{L}_{\ell 1}$ or $\mathcal{L}_{\ell 2}$ (MSE loss), where subspaces are enforced to be *congruent* with the preceding ones, *i.e.*, $f_{\boldsymbol{\theta}}^{n}(\hat{\boldsymbol{x}}) = f_{\boldsymbol{\theta}}^{n-1}(\hat{\boldsymbol{x}})$. Note that in the case of Equ. 8, the orthogonal matrix $\boldsymbol{A}$ is an identity matrix that no rotation and reflection are adopted for simplicity and practicality (more details in Supp.).

4.3. Overall Objective and Algorithm

The overview of our AGD framework to conduct incremental learning is illustrated as Fig. 3. Due to the universal property of our AD mechanism, it is easy to integrate $\mathcal{L}_{\mathcal{G}}^{r\boldsymbol{t}}$ into it. The overall objective is formulated as:

$$\mathcal{L}_{\text{AGD}}(\boldsymbol{x},\hat{\boldsymbol{x}}) = \mathcal{L}_{rep}([\boldsymbol{x}\|\hat{\boldsymbol{x}}]) + \lambda\mathcal{L}_{\mathcal{AD}}(\hat{\boldsymbol{x}};\mathcal{L}_{\mathcal{G}}^{r\boldsymbol{t}}), \tag{9}$$

and optimization procedures are summarized below.

Algorithm 1 Augmented Geometric Distillation (n-th task)

Input: Incremental dataset $\mathcal{D}_{T_n}$ and fixed base model $f_{\boldsymbol{\theta}}^{n-1}$.
Output: Converged evolving model $f_{\boldsymbol{\theta}}^{n}$.

```
Generate dreaming memory M_{T_{1:n-1}} with f_θ^{n-1}.
Initialize the evolving model f_θ^n with f_θ^{n-1}.
while not converged do
    Sample and augment x ⊂ D_{T_n} → x.
    Sample and augment twice x̂ ⊂ M_{T_{1:n-1}} → x̂', x̂''.
    Calculate L_rep (Equ. 1) with f_θ^n(x), f_θ^n(x̂') and f_θ^n(x̂'').
    Calculate L_AD(·; L_G^{rt}) (Equ. 3 and Equ. 8) between
        f_θ^{n-1}(x̂'), f_θ^{n-1}(x̂'') and f_θ^n(x̂'), f_θ^n(x̂'').
    Calculate L_AGD (Equ. 9) and backward.
    Update θ in f_θ^n.
end while
Fix the evolving model f_θ^n for the next step as base model.
```

5. Experiments

5.1. Datasets and Evaluation Protocol

Market-1501 [51] contains 32,668 annotated images of 1,501 identities collected from 6 cameras totally. 12,936 images of 751 identities and 19,732 gallery images are used for training and test respectively.

PersonX [38] is a dataset generated by Unity under controllable cameras and environment. It has 9,840 images / 410 IDs for training and 35,952 images / 856 IDs for test.

Motivation		Method	MSMT17 → Market (M-to-M)						MSMT17 → PersonX (M-to-P)					
			MSMT17		Market		**AVG**		MSMT17		PersonX		**AVG**	
			mAP	Rank-1	mAP	Rank-1	mAP	Rank-1	mAP	Rank-1	mAP	Rank-1	mAP	Rank-1
Oracle		Base Dataset	45.7	71.5	21.8	45.0	33.7	58.3	45.8	71.4	28.0	54.2	36.9	62.8
		Incremental Dataset	2.8	9.2	78.1	90.3	40.5	49.8	1.2	3.6	83.6	93.3	42.4	48.5
Finetune		origin lr	4.8	14.2	<u>81.2</u>	91.8	43.0	53.0	3.1	9.2	**83.9**	**94.1**	43.5	51.7
		1/10 lr	11.0	27.7	78.1	90.9	44.5	59.3	8.3	22.1	81.5	92.9	44.9	57.5
Regularization		EWC [23]	23.2	48.1	66.0	84.1	44.6	66.1	20.9	44.8	61.7	82.0	41.3	63.4
		MAS [2]	22.4	46.7	67.7	85.0	45.1	65.9	22.2	46.6	62.2	82.0	42.2	64.3
Distillation		LwF [24]	9.6	23.4	69.9	85.6	39.7	54.5	5.5	14.2	71.4	83.5	38.5	48.8
		AKA [32]	11.3	27.8	79.5	91.6	45.4	59.7	12.0	18.3	81.6	92.0	46.8	55.2
	Replay [47]	iCaRL [33]	27.6	50.7	**82.8**	**92.8**	55.2	71.8	29.8	53.6	<u>83.4</u>	<u>93.2</u>	56.6	73.4
		ABD [37]	38.5	63.5	79.7	<u>92.0</u>	59.1	77.7	38.9	64.5	79.2	91.3	59.0	77.9
		LUCIR (w/ *cos*) [19]	37.4	62.4	80.4	<u>92.0</u>	58.9	77.2	38.8	64.0	80.9	91.9	<u>59.8</u>	77.9
		LUCIR (w/ ℓ_1) [19]	39.7	65.3	77.8	90.8	58.8	78.1	40.8	66.0	75.7	89.7	58.2	77.9
		LUCIR (w/ ℓ_2) [19]	37.9	63.0	80.2	91.9	59.0	77.5	38.9	64.1	80.5	91.8	59.7	78.0
		PODNet [10]	<u>40.8</u>	<u>66.6</u>	78.3	90.9	<u>59.6</u>	<u>78.7</u>	<u>41.6</u>	<u>67.0</u>	77.7	90.1	59.6	<u>78.6</u>
		GeoDL [36]	38.3	63.7	79.0	91.5	58.7	77.6	39.4	64.6	79.0	91.4	59.2	78.0
		AGD	**41.9**	**67.5**	80.5	91.9	**61.2**	**79.7**	**41.8**	**67.4**	81.0	92.1	**61.4**	**79.9**
Oracle		Joint	48.7	73.7	82.3	92.2	65.5	83.0	46.1	71.6	82.0	92.6	64.0	82.1

Table 1. Comparison with mainstream families of methods in CIL. iCaRL [33] and LUCIR [19]: baseline solutions with Equ. 2 $\kappa = kl$ (Equ. 4) and $\kappa = cos$ (Equ. 5) respectively. Oracle: training with supervision on according dataset(s). **Note that** all results are obtained on *joint gallery* (detailed in Sec. 5.1). For fair comparison, based on the basic representation loss in ReID, we only reproduced distillation parts of *Distillation-based* methods and tuned hyper-parameters for best performance. **Bold** and <u>underline</u>: best and second-best results.

MSMT17 [43] consists of 126,441 bounding boxes of 4,101 identities, of which 32,621 images of 1,041 identities form training set and the remaining form test set.

Evaluation Protocol. After learning incrementally, we denote all test sets of seen tasks as $\mathbb{T} = \{(Q_i, G_i)\}$, where (Q_i, G_i) is the query set and gallery set of the *i*-th task. To evaluate the performance of model in all domains, we define the joint gallery as the intersection of all individual gallery sets, *i.e.*, $\mathcal{G} = \cup_{(Q_i,G_i)\in\mathbb{T}} G_i$, and evaluate each query set in $\mathcal{G}$. Finally, we take average performance as the overall results, *i.e.*, AVG $= {}^1\!/_{|\mathbb{T}|} \sum_{(Q_i,G_i)\in\mathbb{T}} eval(Q_i, \mathcal{G})$, where $eval(\cdot,\cdot)$ outputs mean Average Precision (mAP) and Cumulated Matching Characteristics (CMC) curve as metrics.

5.2. Implementation Details

Following the baseline BoT [27] in ReID, we employ ResNet50 [16], initialized with parameters pre-trained on ImageNet [35], as our backbone. REA [52] (`sh=0.4`), BNNeck [27] are adopted in all training process. Note that stride trick [27] is abandoned for fast training and inference. During inference stage, features after BNNeck will be extracted for final ranking. SGD with learning rate of 0.01 is leveraged to update the parameters. We train the first base model $f^1_{\boldsymbol{\theta}}(\cdot)$ for 90 epochs with warmup and decay learning rate at epoch 61. For incremental tasks, optimization lasts 80 epochs and decay occurs at epoch 41. We generate the dreaming memory until all classes have 40 exemplars or $|\mathcal{M}|$ reaches 40960. When learning incrementally, batch-size is 128, 64 (16 identities $\times$ 4 samples) from $\mathcal{D}_{T_n}$ and $\mathcal{M}_{T_{1:n-1}}$ respectively. Settings of hyper-parameters are detailed in Sec. 5.4.

5.3. Comparison with Other Methods

After replacing the memory built by real preceding data with dreaming data, we adapt typical methods in CIL to ReID for comparison as summarized in Tab. 1. We will analyze results mainly on MSMT17 → Market (M-to-M) and take AVG as overall performance.

Oracle: All results in "Oracle" family are achieved under supervised training protocol. As expected, both base and incremental tasks achieve satisfactory results after "Joint" training.

Finetune: Similar to results in CIL and AKA [32], finetuning directly induces catastrophic forgetting in base task. A vanilla approach to alleviate it is decreasing the finetuning learning rate. Despite the sacrifice of performance on incremental dataset, "1/10 lr" still yields better overall performance (1.5% mAP / 6.3% R@1) over "origin lr".

Regularization: EWC [23] and MAS [2] constrain updating important parameters explicitly to balance knowledge learning and retaining. The results demonstrate its effectiveness in mitigating forgetting. On the other side, the explicit penalization of parameter updating disturbs the fitting on incremental dataset dramatically (10.4+% mAP / 5.9+% R@1 compared with "Finetune").

Distillation and Ours: Distillation-based methods intend to transfer knowledge from the base model to the evolving model to combat forgetting. This family of methods acts as the mainstream and leads the SOTAs in CIL. LwF [24] and iCaRL [33] focus on the distribution on preceding classes. With dreaming data as memory, iCaRL outperforms LwF greatly (15.5% mAP / 17.3% R@1). AKA [32] leverages a graph to manage knowledge. How-

Method	MSMT17 → Market (M-to-M)							
	MSMT17		Market		**AVG**			
	mAP	R@1	mAP	R@1	mAP		R@1	
iCaRL [33]	27.6	50.7	82.8	92.8	55.2	+0.0	71.8	+0.0
LUCIR [19]	37.4	62.4	80.4	92.0	58.9	+0.0	77.2	+0.0
w/ AD	30.4	54.3	83.5	92.7	57.0	+1.8	73.5	+1.7
w/ AD	39.1	64.6	80.4	91.7	59.7	+0.8	78.2	+1.0
w/ $\mathcal{L}_{\mathcal{G}}^{r}$	39.0	64.8	79.8	91.7	59.4	+0.5	78.3	+1.1
w/ $\mathcal{L}_{\mathcal{G}}^{r\boldsymbol{t}}$	39.8	65.5	80.4	91.6	60.1	+1.2	78.5	+1.3
w/o $\mathcal{M}$	23.5	46.0	40.6	67.0	32.0	-26.9	56.5	-20.7
w/ $\mathcal{L}_{\mathcal{G}}^{r\boldsymbol{t}}$ (bs x2)	41.2	66.8	78.7	90.6	60.0	+1.1	78.7	+1.5
w/ $\mathcal{L}_{\mathcal{G}}^{r\boldsymbol{t}}$ (ep x2)	38.5	64.9	81.7	92.2	60.1	+1.2	78.6	+1.4
w/ $\mathcal{L}_{\mathcal{G}}^{r\boldsymbol{t}}$ (re /2)	40.5	66.2	80.3	91.9	60.4	+1.5	79.1	+1.9
AGD	**41.9**	**67.5**	**80.5**	**91.9**	**61.2**	+6.0 +2.3	**79.7**	+7.9 +2.5

Table 2. Ablation studies. iCaRL (Equ. 2 $\kappa = kl$) and LUCIR (Equ. 2 $\kappa = cos$) serve as baselines. "bs x2": with larger batch size, *i.e.*, 256(=128+128). "ep x2": train longer, *i.e.*, 160 epochs. "re /2": erasing less area of dreaming data (weak augmentation). Comparisons are marked in colors (blue: comparisons with iCaRL, green: comparisons with LUCIR).

ever, absence of memory inhibits its performance. In replay family, LUCIR [19] and ABD [37] demonstrate advantage of distillation on features with the improvements of 3.6+% mAP / 5.4+% R@1 over iCaRL. And PODNet [10] additionally penalizes the drift of intermediate attention maps, which brings another 0.7% mAP / 1.5% R@1 gain. Attempting to transfer more knowledge, GeoDL [36] proposes to distill geodesic flow and achieves 0.4% R@1 boost over LUCIR. Our method relies on augmented distillation to enhance effectiveness of low quality dreaming memory. Besides, geometric distillation memorizes relative information, which is critical for retrieval task and meanwhile keeps flexible and plastic for incremental tasks. Combined, ours yields results of 61.2% mAP / 79.7% R@1, which surpasses other methods by a margin on AVG performance (1.6+% mAP / 1.0+% R@1). It is noteworthy that without attention maps, our method only puts constrain on the final features, which indicates its great advance in effectiveness.

5.4. Ablation Studies and Parameter Analysis

In this section, we perform ablation studies and parameter analysis to investigate the contribution of each component in AGD to the final performance gain and evaluations on different settings. Results are shown in Tab. 2.

Effectiveness of Augmented Distillation. Augmented distillation aims at mitigating "noisy distillation", particularly when driven by dreaming exemplars. Based on both baselines iCaRL and LUCIR, which transfer knowledge from two different perspectives (detailed in Equ. 4 and Equ. 5 in Sec. 4.1), our proposal brings gain of 1.8% mAP / 1.7% R@1 and 0.8% mAP / 1.0% R@1 respectively. When incorporated into our $\mathcal{L}_{\mathcal{G}}^{r\boldsymbol{t}}$, "AGD" outperforms "w/ $\mathcal{L}_{\mathcal{G}}^{r\boldsymbol{t}}$" with a margin of 1.1% mAP and 1.2% R@1. The consistent improvements demonstrate its generalization on different distillation terms. To further investigate the rationale of AD mechanism, we train the networks with larger batch size, longer period and weak augmentation for dreaming data, which are the empirical approaches to stabilize training. However, "w/ $\mathcal{L}_{\mathcal{G}}^{r\boldsymbol{t}}$ (bs x2)" and "w/ $\mathcal{L}_{\mathcal{G}}^{r\boldsymbol{t}}$ (ep x2)" both fail to surpass "w/ $\mathcal{L}_{\mathcal{G}}^{r\boldsymbol{t}}$". "w/ $\mathcal{L}_{\mathcal{G}}^{r\boldsymbol{t}}$ (re /2)" achieves marginal gain but is not capable to defeat "AGD". The fact indicates that different from increasing batch size or training longer directly, AD mechanism digs more information in noisy exemplars effectively without hurting knowledge learning in incremental domain. And this is of great advances for such data-limited scenario as incremental learning.

Effectiveness of Geometric Distillation. Casting the distillation term on features is verified to be crucial as aforementioned. To further investigate the necessity of geometric distillation, we adopt $\mathcal{L}_{cos}$, $\mathcal{L}_{\ell 1}$ and $\mathcal{L}_{\ell 2}$ (MSE loss) to conduct extensive experiments. $\mathcal{L}_{cos}$ requires input pair-wise features to have the same orientations, while $\mathcal{L}_{\ell 1}$ and $\mathcal{L}_{\ell 2}$ enforce the features to remain unchanged, *i.e.*, the preceding feature space is congruent after evolving (the bijection function is $f_{\boldsymbol{\theta}}^{n}(\boldsymbol{x}) = f_{\boldsymbol{\theta}}^{n-1}(\boldsymbol{x})$). After tuning weighting parameter λ, $\mathcal{L}_{cos}$, $\mathcal{L}_{\ell 1}$ and $\mathcal{L}_{\ell 2}$ all yield the satisfactory performance on M-to-M task (Tab. 1). But when compared with $\mathcal{L}_{\mathcal{G}}^{r}$ (Tab. 2), 0.5% mAP / 1.1% R@1 decreases are shown. When we allow more necessary drift, $\mathcal{L}_{\mathcal{G}}^{r\boldsymbol{t}}$ achieves another 0.7% mAP / 0.2% R@1 improvements and surpasses LUCIR with advances of 1.2% mAP / 1.3% R@1. The gain justifies the superiority of geometric distillation, which makes our framework flexible yet retentive.

Effectiveness of Dreaming Memory. In our framework, the distillation term is completely driven by dreaming memory $\mathcal{M}$. In addition to the privacy issue, it plays a central role in building the similarity feature subspaces. In Tab. 1, replay-based family of methods surpasses others with a huge margin, which validates the necessity of dreaming memory. To further measure its contribution, we remove $\mathcal{M}$ and cast the distillation term ($\mathcal{L}_{cos}$) directly on incremental dataset $\mathcal{D}_{T_n}$. An serious degradation of 26.9% mAP / 20.7% R@1 is observed in Tab. 2, especially in incremental domain, which completely ruins the results. This shows that $\mathcal{M}$ decouples the objectives of learning and reviewing, avoiding the potential interference.

Evaluation on α. α determines the weight of cross part in AD (Equ. 3). According to the curve in Fig. 6, "$\alpha = 0.9$" performs best, which demonstrates necessity of multi-view guidance for robust feature distillation.

Evaluation on λ. λ is the weight factor of overall distillation term. Larger weight leads to less forget and less flexibility. Relatively, our framework is not sensitive to λ and "$\lambda = 3$" yields the best results.

Evaluation on peers. *peers*, *i.e.*, number of views in each distillation iteration, is fixed to 2 by default. A larger *peers* will provides guidance from more views and stronger

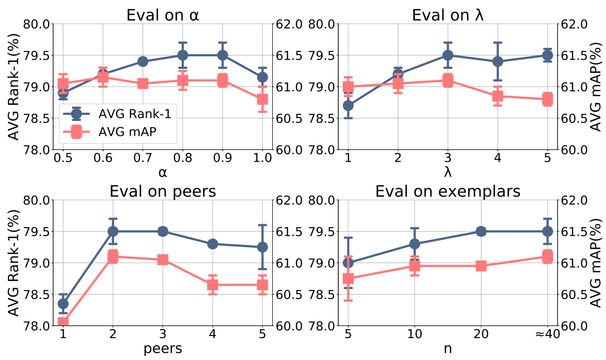

Figure 6. Evaluations on different settings (MSMT17 $\rightarrow$ Market). **Top Left:** Evaluation on α. **Top Right:** Evaluation on λ. **Bottom Left:** Evaluation on peers. **Bottom Right:** Evaluation on exemplars per ID. Typically, $\alpha = 0.9$, $\lambda = 3$, $peers = 2$ and $n \approx 40$.

Method	MSMT17 $\rightarrow$ Market $\rightarrow$ PersonX (M-to-M-to-P)							
	MSMT17		Market		PersonX		**AVG**	
	mAP	R@1	mAP	R@1	mAP	R@1	mAP	R@1
Oracle-Joint	48.0	73.1	82.3	93.2	83.9	93.4	71.4	86.6
Finetune	1.5	4.9	15.0	36.8	**85.0**	**93.8**	33.8	45.2
iCaRL [33]	16.4	35.2	66.0	83.5	84.9	93.3	55.7	70.6
LUCIR [19]	28.3	52.5	68.5	85.8	82.6	93.0	59.8	77.1
AGD	**36.5**	**62.4**	**71.9**	**87.3**	83.6	93.5	**64.0**	**81.0**

Method	MSMT17 $\rightarrow$ PersonX $\rightarrow$ Market (M-to-P-to-M)							
	MSMT17		PersonX		Market		**AVG**	
	mAP	R@1	mAP	R@1	mAP	R@1	mAP	R@1
Oracle-Joint	48.0	73.1	83.9	93.4	82.3	93.2	71.4	86.6
Finetune	2.7	8.1	24.8	48.3	81.7	**92.5**	36.4	49.6
iCaRL [33]	16.7	35.1	59.3	75.7	**82.6**	92.3	52.9	67.7
LUCIR [19]	27.2	50.7	62.9	80.1	79.5	91.6	56.5	74.1
AGD	**36.4**	**61.9**	**67.4**	**83.4**	80.5	91.6	**61.4**	**78.9**

Table 3. Extensive experiments under more tasks settings.

regularization. However, as shown in Fig. 6 (Bottom Left), no extra gain is observed and we think it is because that average of guidances from too many views weakens the diversity in each view and over-regularizes the distillation.

Evaluation on exemplars. In general, more exemplars report better performance due to the more diversity of memory. In our framework, "$\approx$ 40 exemplar" (40960 in total / 1041 IDs in MSMT17) outperforms other settings. But it is noteworthy that much less exemplars only result in about 0.5% degradation, which confirms the effectiveness of our method from the other side.

5.5. Further Discussion

Learning More Tasks. When learning incrementally with more tasks, "Finetune" performs similarly that encounters catastrophic forgetting. Dreaming memory alleviates such forgetting to a great extent, which brings iCaRL [33] and LUCIR [19] the huge gain. Furthermore, our AGD makes more advantage of $\mathcal{M}$ and yields compelling improvements of 25.0+% mAP / 29.3+% R@1 over "Finetune" on both tasks settings.

Method	CIFAR100 (20 exemplars per class in $\mathcal{M}$)							
	50 steps		25 steps		10 steps		5 steps	
	Inc Acc		Inc Acc		Inc Acc		Inc Acc	
LUCIR [19]	54.6	+0.0	61.3	+0.0	63.4	+0.0	65.1	+0.0
w/ $\mathcal{L}_{\mathcal{G}}^{rt}$	59.2	+4.6	62.1	+0.8	64.1	+0.7	65.3	+0.2
w/ AGD	60.7	+6.1	62.6	+1.3	64.6	+1.2	65.5	+0.4
PODNet [10]	61.5	+0.0	63.3	+0.0	64.4	+0.0	65.3	+0.0
w/ $\mathcal{L}_{\mathcal{G}}^{rt}$	62.5	+1.0	64.2	+0.9	65.0	+0.6	65.6	+0.3
w/ AGD	62.9	+1.4	64.3	+1.0	65.3	+0.9	65.7	+0.4

Table 4. Extensive experiments on CIL (CIFAR100). 50 classes for pre-training and 50 classes for incremental tasks.

Class Incremental Learning with NME. To investigate whether CIL could benefit from our AGD, we execute extensive experiments on CIFAR100. The *nearest-mean-of-exemplars* (NME) [33] rule is adopted as classifier to better meet the scenario of retrieval task and *Average Incremental Accuracy* is the evaluation metric (detailed in Supp.). Combined with LUCIR [19], both parts of our AGD improve the accuracy, especially in "50 steps" setting, 6.1% acc gain is shown. Even incorporated with a stronger solution (one of the SOTAs) PODNet [10], AGD performs well and achieves 62.9% acc in the most challenging "50 steps" setting. The results suggest that in CIL, preserving the structure of feature space when evolving could be beneficial, despite the fact that CIL aims at classification, not ranking. Another impressive thing is that in experiments, $\mathcal{M}$ stores 20 exemplars each class for replaying and these exemplars are in real image distribution exactly. And we believe AD mechanism works here mainly because that artificial image augmentations introduce some noise into geometric distillation, which focuses on the residual vectors of features and is more sensitive to the noise in features.

6. Conclusion

In this work, we have developed the AGD framework, which is an incremental framework tailored-made for ReID. It replays preceding knowledge via dreaming memory without privacy issue, and augments the "noisy distillation" in a novel crisscross pattern, uncovering the potential information in dreaming memory from noise. Moreover, we have stricken a better balance between learning and memorizing in a geometric way, where semantic drift is allowed to adapt new knowledge and preceding knowledge is preserved via maintaining space structure when drifting. Finally, superiority to typical solutions in CIL validates its promising potential when adopted in ReID, open-set incremental tasks and even more conventional CIL.

Acknowledgement

This work was partially supported by the National Natural Science Foundation of China under Grants No. 61871052 and 62192784 and Alibaba Group through Alibaba Innovation Research Program.

References

[1] Davide Abati, Jakub Tomczak, Tijmen Blankevoort, Simone Calderara, Rita Cucchiara, and Babak Ehteshami Bejnordi. Conditional channel gated networks for task-aware continual learning. In *CVPR*, pages 3931–3940, 2020. 2

[2] Rahaf Aljundi, Francesca Babiloni, Mohamed Elhoseiny, Marcus Rohrbach, and Tinne Tuytelaars. Memory aware synapses: Learning what (not) to forget. In *ECCV*, pages 139–154, 2018. 2, 6

[3] Kartikeya Bhardwaj, Naveen Suda, and Radu Marculescu. Dream distillation: A data-independent model compression framework. In *ICML Worshops*, 2019. 2

[4] Yaohui Cai, Zhewei Yao, Zhen Dong, Amir Gholami, Michael W Mahoney, and Kurt Keutzer. Zeroq: A novel zero shot quantization framework. In *CVPR*, pages 13169–13178, 2020. 2

[5] Francisco M Castro, Manuel J Marín-Jiménez, Nicolás Guil, Cordelia Schmid, and Karteek Alahari. End-to-end incremental learning. In *ECCV*, pages 233–248, 2018. 1, 2

[6] Hanting Chen, Yunhe Wang, Chang Xu, Zhaohui Yang, Chuanjian Liu, Boxin Shi, Chunjing Xu, Chao Xu, and Qi Tian. Data-free learning of student networks. In *ICCV*, pages 3514–3522, 2019. 2

[7] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, pages 1597–1607. PMLR, 2020. 3

[8] Yoojin Choi, Jihwan Choi, Mostafa El-Khamy, and Jungwon Lee. Data-free network quantization with adversarial knowledge distillation. In *CVPR Workshops*, pages 710–711, 2020. 2

[9] Prithviraj Dhar, Rajat Vikram Singh, Kuan-Chuan Peng, Ziyan Wu, and Rama Chellappa. Learning without memorizing. In *CVPR*, pages 5138–5146, 2019. 2

[10] Arthur Douillard, Matthieu Cord, Charles Ollion, Thomas Robert, and Eduardo Valle. Podnet: Pooled outputs distillation for small-tasks incremental learning. In *ECCV*, pages 86–102. Springer, 2020. 1, 2, 6, 7, 8

[11] Gongfan Fang, Jie Song, Chengchao Shen, Xinchao Wang, Da Chen, and Mingli Song. Data-free adversarial distillation. *arXiv preprint arXiv:1912.11006*, 2019. 2

[12] Robert M French. Catastrophic forgetting in connectionist networks. *Trends in cognitive sciences*, 3(4):128–135, 1999. 1, 2

[13] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, Bilal Piot, koray kavukcuoglu, Remi Munos, and Michal Valko. Bootstrap your own latent - a new approach to self-supervised learning. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *NeurIPS*, volume 33, pages 21271–21284. Curran Associates, Inc., 2020. 3

[14] Matan Haroush, Itay Hubara, Elad Hoffer, and Daniel Soudry. The knowledge within: Methods for data-free model compression. In *CVPR*, pages 8494–8502, 2020. 2

[15] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, pages 9729–9738, 2020. 3

[16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 6

[17] Geoffrey Hinton, Oriol Vinyals, and Jeffrey Dean. Distilling the knowledge in a neural network. In *NeurIPS Workshops*, 2015. 2

[18] Saihui Hou, Xinyu Pan, Chen Change Loy, Zilei Wang, and Dahua Lin. Lifelong learning via progressive distillation and retrospection. In *ECCV*, pages 437–452, 2018. 1, 2

[19] Saihui Hou, Xinyu Pan, Chen Change Loy, Zilei Wang, and Dahua Lin. Learning a unified classifier incrementally via rebalancing. In *CVPR*, pages 831–839, 2019. 1, 2, 4, 6, 7, 8

[20] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, pages 448–456. PMLR, 2015. 2

[21] Xu Ji, João Henriques, Tinne Tuytelaars, and Andrea Vedaldi. Automatic recall machines: Internal replay, continual learning and the brain. In *NeurIPS Workshops*. NeurIPS, 2020. 2

[22] Menelaos Kanakis, David Bruggemann, Suman Saha, Stamatios Georgoulis, Anton Obukhov, and Luc Van Gool. Reparameterizing convolutions for incremental multi-task learning without task interference. In *ECCV*, pages 689–707. Springer, 2020. 2

[23] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *PNAS*, 114(13):3521–3526, 2017. 2, 6

[24] Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE TPAMI*, 40(12):2935–2947, 2017. 1, 2, 6

[25] Yufan Liu, Jiajiong Cao, Bing Li, Chunfeng Yuan, Weiming Hu, Yangxi Li, and Yunqiang Duan. Knowledge distillation via instance relationship graph. In *CVPR*, pages 7096–7104, 2019. 2

[26] Raphael Gontijo Lopes, Stefano Fenu, and Thad Starner. Data-free knowledge distillation for deep neural networks. In *NeurIPS Workshops*, 2017. 2

[27] Hao Luo, Wei Jiang, Youzhi Gu, Fuxu Liu, Xingyu Liao, Shenqi Lai, and Jianyang Gu. A strong baseline and batch normalization neck for deep person re-identification. *TMM*, 2019. 6

[28] Arun Mallya, Dillon Davis, and Svetlana Lazebnik. Piggyback: Adapting a single network to multiple tasks by learning to mask weights. In *ECCV*, pages 67–82, 2018. 2

[29] Michael McCloskey and Neal J Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of learning and motivation*, volume 24, pages 109–165. Elsevier, 1989. 1

[30] Paul Micaelli and Amos J Storkey. Zero-shot knowledge transfer via adversarial belief matching. In *NeurIPS*, pages 9551–9561, 2019. 2

[31] Ameya Prabhu, Philip HS Torr, and Puneet K Dokania. Gdumb: A simple approach that questions our progress in

continual learning. In *ECCV*, pages 524–540. Springer, 2020. 2

[32] Nan Pu, Wei Chen, Yu Liu, Erwin M Bakker, and Michael S Lew. Lifelong person re-identification via adaptive knowledge accumulation. In *CVPR*, pages 7901–7910, 2021. 1, 6

[33] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. icarl: Incremental classifier and representation learning. In *CVPR*, pages 2001–2010, 2017. 1, 2, 4, 6, 7, 8

[34] Anthony Robins. Catastrophic forgetting, rehearsal and pseudorehearsal. *Connection Science*, 7(2):123–146, 1995. 2

[35] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *IJCV*, 115(3):211–252, 2015. 6

[36] Christian Simon, Piotr Koniusz, and Mehrtash Harandi. On learning the geodesic path for incremental learning. In *CVPR*, pages 1591–1600, 2021. 2, 6, 7

[37] James Smith, Yen-Chang Hsu, Jonathan Balloch, Yilin Shen, Hongxia Jin, and Zsolt Kira. Always be dreaming: A new approach for data-free class-incremental learning. In *ICCV*, pages 9374–9384, October 2021. 1, 2, 6, 7

[38] Xiaoxiao Sun and Liang Zheng. Dissecting person re-identification from the viewpoint of viewpoint. In *CVPR*, 2019. 5

[39] Xiaoyu Tao, Xinyuan Chang, Xiaopeng Hong, Xing Wei, and Yihong Gong. Topology-preserving class-incremental learning. In *ECCV*, pages 254–270. Springer, 2020. 2

[40] Xiaoyu Tao, Xiaopeng Hong, Xinyuan Chang, Songlin Dong, Xing Wei, and Yihong Gong. Few-shot class-incremental learning. In *CVPR*, pages 12183–12192, 2020. 2

[41] Sebastian Thrun. Lifelong learning algorithms. In *Learning to learn*, pages 181–209. Springer, 1998. 2

[42] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008. 4

[43] Longhui Wei, Shiliang Zhang, Wen Gao, and Qi Tian. Person transfer gan to bridge domain gap for person re-identification. In *CVPR*, pages 79–88, 2018. 6

[44] Yue Wu, Yinpeng Chen, Lijuan Wang, Yuancheng Ye, Zicheng Liu, Yandong Guo, and Yun Fu. Large scale incremental learning. In *CVPR*, pages 374–382, 2019. 1, 2

[45] Guodong Xu, Ziwei Liu, Xiaoxiao Li, and Chen Change Loy. Knowledge distillation meets self-supervision. In *ECCV*, pages 588–604. Springer, 2020. 2

[46] Junho Yim, Donggyu Joo, Jihoon Bae, and Junmo Kim. A gift from knowledge distillation: Fast optimization, network minimization and transfer learning. In *CVPR*, pages 4133–4141, 2017. 2

[47] Hongxu Yin, Pavlo Molchanov, Jose M Alvarez, Zhizhong Li, Arun Mallya, Derek Hoiem, Niraj K Jha, and Jan Kautz. Dreaming to distill: Data-free knowledge transfer via deepinversion. In *CVPR*, pages 8715–8724, 2020. 1, 2, 3, 6

[48] Jaemin Yoo, Minyong Cho, Taebum Kim, and U Kang. Knowledge extraction with no observable data. In *NeurIPS*, pages 2705–2714, 2019. 2

[49] Lu Yu, Bartlomiej Twardowski, Xialei Liu, Luis Herranz, Kai Wang, Yongmei Cheng, Shangling Jui, and Joost van de Weijer. Semantic drift compensation for class-incremental learning. In *CVPR*, pages 6982–6991, 2020. 1, 2

[50] Sergey Zagoruyko and Nikos Komodakis. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. In *ICLR*, 2016. 2

[51] Liang Zheng, Liyue Shen, Lu Tian, Shengjin Wang, Jingdong Wang, and Qi Tian. Scalable person re-identification: A benchmark. In *ICCV*, pages 1116–1124, 2015. 5

[52] Zhun Zhong, Liang Zheng, Guoliang Kang, Shaozi Li, and Yi Yang. Random erasing data augmentation. In *AAAI*, volume 34, pages 13001–13008, 2020. 3, 6