DK-DDIL: Adaptive Knowledge Retention for Dynamic Domain-Incremental Learning in Medical Imaging

Supplementary Material

The *Supplementary Material* provides additional theoretical analysis, implementation details, and extended experimental results that further support the findings presented in the main paper. Its organization is as follows: Sec. 6 elaborates on the theoretical motivation underlying the adaptive fusion scheduling strategy employed in MKI. Sec. 7 outlines the complete algorithmic pipeline of the proposed DK-DDIL framework. Sec. 8 provides an expanded analysis of domain-order effects, examining how different domain permutations influence the performance of various methods. Sec. 9 presents an ablation study on the DCL objective and analyzes the contribution of each loss component. Sec. 10 details the complete experimental setup, including dataset specifications, implementation settings, evaluation metrics, the extension to 3D MRI scenarios, and clarifications on the replay configurations adopted in our experiments.

6. Theoretical Motivation of Adaptive Fusion Scheduling in MKI

The MKI module regulates the interpolation between inherited and newly optimized parameters through a dynamic fusion coefficient α_e. Conceptually, α_e mediates the trade-off between *stability* (retention of previously acquired knowledge) and *plasticity* (adaptation to new domain data), which is central to dynamic DIL. From a theoretical perspective, inappropriate scheduling of α_e can be detrimental: overly aggressive decay prematurely suppresses knowledge retention, leading to *catastrophic forgetting*, whereas overly conservative decay limits adaptation, reducing the model's domain-specific performance.

We formalize the temporal evolution of knowledge inheritance as a smooth, differentiable, and monotonically decreasing function $\alpha_e : [0, E] \rightarrow [\alpha_{\text{final}}, \alpha_{\text{init}}]$ over training epochs $e \in \{0, \ldots, E\}$. The chosen cosine-annealed schedule satisfies two key theoretical desiderata:

1. **First-order continuity.** The trajectory of α_e is continuously differentiable, ensuring that $\frac{d\alpha_e}{de}$ varies smoothly across epochs. This avoids abrupt parameter updates and maintains stable inter-domain transitions.
2. **Zero-gradient endpoints.** At $e = 0$ and $e = E$, the derivative $\frac{d\alpha_e}{de} = 0$, which guarantees that early-stage inherited knowledge is not instantaneously overwritten and that the fusion process converges to a stable fixed point at the final epoch.

Formally, the cosine-annealed schedule is expressed as Eq. 16. This design guarantees a theoretically smooth interpolation from stability-dominant to plasticity-dominant regimes.

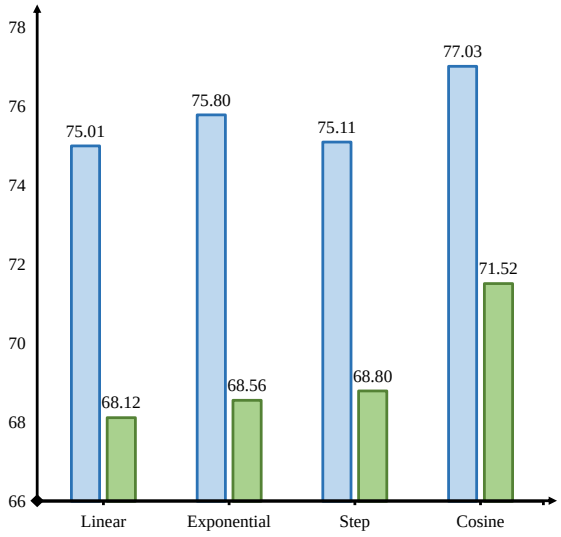

Figure 4. Comparison of DK-DDIL performance using **four decay strategies** on the **Skin Pathology Diagnosis** dataset. Blue and green bars represent $\overline{A}$ and A_T, respectively.

We briefly contrast cosine-annealing with widely used decay strategies:

- **Linear Decay** produces constant-rate reduction but exhibits discontinuous derivatives at boundary points, which can induce non-smooth fusion updates and unstable domain transitions.
- **Exponential Decay** ensures monotonicity but decays excessively rapidly in early stages, prematurely weakening inherited knowledge and yielding a suboptimal balance between stability and plasticity.
- **Step Decay** introduces discrete, large-magnitude drops in α_e. Such discontinuities conflict with the continuous nature of domain evolution and often lead to optimization shocks.

Empirical evaluation on the **Skin Pathology Diagnosis** dataset (Fig. 4) demonstrates that cosine-annealed scheduling consistently outperforms Linear, Exponential, and Step decays in both $\overline{A}$ and A_T, confirming its superior stability–plasticity balance. These results validate that cosine-annealed fusion achieves both theoretical rigor and empirical robustness, establishing it as an effective design choice for MKI in DK-DDIL.

7. Algorithmic Overview of DK-DDIL

To provide a clear view of DK-DDIL, we summarize its overall training procedure in Algorithm 1. The algorithm proceeds sequentially over incoming domains with evolving label spaces. For each domain, the model first performs parameter adaptation via the DAM, followed by representational refinement through the KIR mechanism, which includes both MKI and DCL.

8. Influence of Domain Ordering

Domain ordering is a key factor in DIL, as the sequence in which domains arrive can significantly shape the optimization trajectory and the dynamics of knowledge retention. Early domains often exert disproportionate influence on the backbone representation, while later domains introduce distributional shifts that challenge the model's plasticity. Consequently, methods without explicit mechanisms for mitigating order-induced bias typically exhibit substantial performance variability across different domain permutations.

Table 2 reports the results of state-of-the-art DIL methods under four permutations of the OfficeHome domains. We observe that most baselines experience fluctuations in both average accuracy $\overline{A}$ and final accuracy A_T. This sensitivity highlights their reliance on domain-specific cues and their limited capacity to maintain stable knowledge trajectories when the order of domain exposure changes. In particular, prompt-based approaches and rehearsal-free adaptation strategies tend to drift more severely when early domains impose strong biases on the shared representation.

In contrast, DK-DDIL achieves consistently high performance across all domain orders. This robustness arises naturally from the design principles of DK-DDIL. First, the DAM regulates LoRA capacity through learnable rank masks and sparsity-aware modulation, allowing later domains to acquire additional expressive power rather than being constrained by earlier subspace allocations. Second, the KIR mechanism ensures that domain-invariant information is selectively carried forward and reconciled with new updates, preventing early domains from overwriting the optimization trajectory. Together, these mechanisms form a stable, order-insensitive adaptation pathway in which each domain is allocated capacity proportional to its complexity while maintaining consistency with previously learned knowledge.

Overall, DK-DDIL exhibits strong cross-order consistency, indicating that its adaptation strategy forms a domain-agnostic and resilient knowledge structure. This property is particularly advantageous in real-world deployment, where the arrival order of domains is uncontrolled and may vary across clinical sites or data streams.

9. Results Analysis of DCL

The ablation study in Fig. 5 evaluates the impact of each component in the DCL framework. The baseline, using only Cross-Entropy loss ($\mathcal{L}_{\text{CE}}$), achieves an $\overline{A}$ performance of 73.47. Adding the positive alignment term ($\mathcal{L}_{\text{pos}}$) improves the performance to 75.60, reflecting the benefit of aligning feature embeddings with class prototypes. However, this does not address domain shifts or inter-class separation, which limits its effectiveness. Introducing the negative contrastive loss ($\mathcal{L}_{\text{neg}}$) boosts the performance to 75.43 by enhancing separation between classes. Adding intra-class compactness ($\mathcal{L}_{\text{intra}}$) results in an $\overline{A}$ score of 74.70, improving feature coherence within the same class but still leaving cross-domain misalignment unaddressed.

Combining $\mathcal{L}_{\text{CE}}$ with both positive alignment and negative contrastive loss ($\mathcal{L}_{\text{pos}}$ + $\mathcal{L}_{\text{neg}}$) leads to a significant performance increase, achieving an A_T score of 69.88. This combination stabilizes feature representations and separates classes more effectively. A similar improvement is achieved by combining positive alignment and intra-class compactness ($\mathcal{L}_{\text{pos}}$ + $\mathcal{L}_{\text{intra}}$), reaching an $\overline{A}$ score of 76.43, which emphasizes the importance of both alignment and feature cohesion. The combination of the negative contrastive losses ($\mathcal{L}_{\text{neg}}$), including intra-domain separation and cross-domain confusion suppression, with intra-class compactness ($\mathcal{L}_{\text{intra}}$), results in an $\overline{A}$ performance of 75.39. This improves separation within domains and reduces cross-domain confusion, but still lacks the alignment with class prototypes.

Finally, the full DCL objective, combining positive alignment, negative intra-domain contrastive loss, cross-domain confusion suppression, and intra-class compactness ($\mathcal{L}_{\text{pos}} + \mathcal{L}_{\text{neg-intra}} + \mathcal{L}_{\text{neg-cross}} + \mathcal{L}_{\text{intra}}$), achieves the best $\overline{A}$ performance of 77.03. This demonstrates the effectiveness of integrating all components. Positive alignment mitigates prototype drift, negative contrastive losses prevent cross-domain confusion, and intra-class compactness improves feature coherence, resulting in a more robust model for dynamic domain-incremental learning. This highlights the importance of combining these components to address the complex challenges of domain misalignment, inter-class separation, and evolving class distributions.

10. Extended Experimental Settings

10.1. Datasets

To evaluate the effectiveness and generalization ability of DK-DDIL, we conduct experiments on three representative scenarios: **Skin Pathology Diagnosis** [5, 11, 13, 21, 22, 30], **Cyst-X** [24], and **OfficeHome** [31].

The **Skin Pathology Diagnosis** setting is constructed from real-world dermoscopic data aggregated from multiple public sources, including PH2 [21], MSK [5], D7P [13],

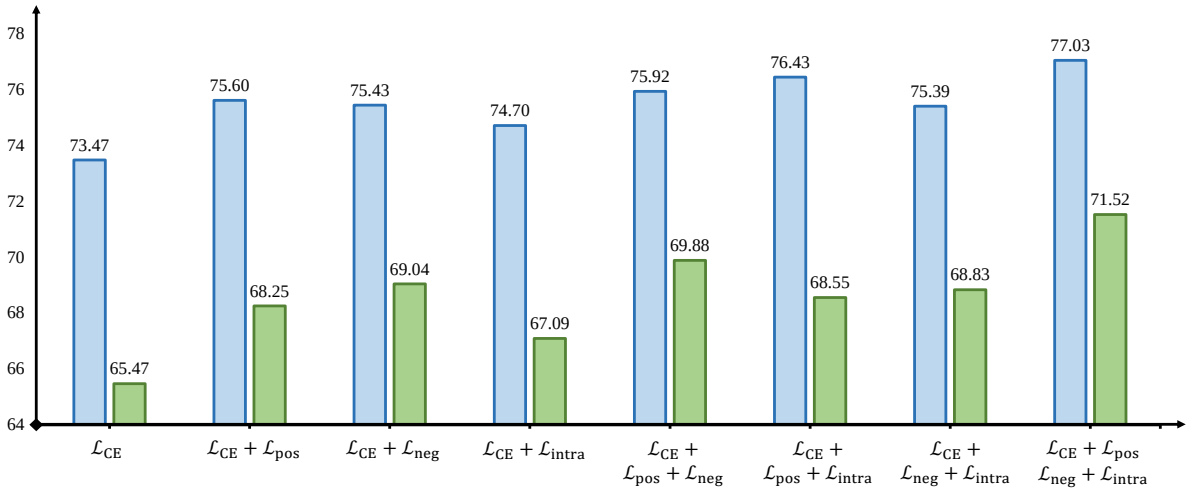

Figure 5. Ablation study on loss components in the DCL module on the **Skin Pathology Diagnosis** dataset. Removing or combining loss terms ($\mathcal{L}_{pos}$, $\mathcal{L}_{neg}$, $\mathcal{L}_{intra}$) demonstrates their individual and joint contributions to model performance. Blue and green bars represent $\overline{A}$ and A_T, respectively.

Table 2. Comparison of state-of-the-art DIL methods on the **OfficeHome** dataset under different domain orders. The four domains — Art (A), Clipart (C), Product (P), and Real-World (R) — are denoted by their initials. We report both the average accuracy $\overline{A}$ and final accuracy A_T. Note that the **A-C-P-R** sequence corresponds to the OfficeHome results reported in Table 1.

Methods	p-**value**	**C-P-R-A**		**P-R-A-C**		**R-A-C-P**	
		$\overline{A}$	A_T	$\overline{A}$	A_T	$\overline{A}$	A_T
Finetune	< 0.01	79.49	78.82	87.28	81.66	83.30	81.36
L2P[34]	< 0.01	77.19	80.72	86.73	80.57	83.05	80.53
DualPrompt[33]	< 0.01	77.52	80.40	87.15	80.77	82.51	79.72
CODA-Prompt[27]	< 0.01	82.48	84.38	89.62	84.31	85.51	83.78
RanPAC[19]	< 0.01	83.73	85.72	89.89	85.57	86.26	85.42
EASE[39]	< 0.01	80.25	83.03	87.68	83.18	84.71	83.27
SimpleCIL[41]	< 0.01	73.12	78.44	84.01	78.44	81.35	78.44
Aper w/ VPT-Shallow[41]	< 0.01	76.25	80.96	86.35	80.96	82.74	80.10
Aper w/ VPT-Deep[41]	< 0.01	78.65	82.52	86.77	81.24	83.46	81.04
Aper w/ SSF[41]	< 0.01	79.75	83.24	87.41	82.11	83.91	81.53
Aper w/ Finetune[41]	< 0.01	55.00	62.26	59.69	53.28	53.79	52.68
Aper w/ Adapter[41]	< 0.01	75.63	80.19	85.37	79.85	82.10	79.35
DGR[9]	< 0.01	73.95	63.11	74.93	65.21	71.03	65.48
GC2[2]	< 0.01	80.16	79.85	87.05	81.79	82.54	80.79
MOS[29]	< 0.01	82.59	85.15	89.36	84.33	84.55	83.61
DUCT[42]	< 0.05	80.27	85.64	90.22	85.57	87.19	85.81
CL-LoRA[10]	< 0.01	79.36	83.78	89.13	83.95	85.75	84.14
DK-DDIL (Ours)	**-**	**85.70**	**87.22**	**91.23**	**86.45**	**87.30**	**85.85**

HAM [30], BCN [11], and other ISIC subsets [22], with the sample distribution shown in Fig. 2. Unlike static benchmarks, this dataset reflects the genuine *temporal evolution* of clinical practice: new imaging devices, centers, and diagnostic categories appear sequentially over time. We organize the data chronologically based on acquisition periods and clinical sources, thereby forming a realistic dynamic DIL scenario. This design captures the real-world challenges of continual adaptation in dermatology, where category composition and imaging distribution evolve with time. These settings jointly cover real-world medical dynamics and standard benchmarks, enabling a comprehensive evaluation of DIL under both clinical and natural conditions.

The **Cyst-X** dataset [24] is a large-scale multi-center 3D MRI collection for Intraductal Papillary Mucinous Neo-

plasm (IPMN) risk stratification. It comprises 1,461 T1-weighted MRI scans from seven medical institutions, covering diverse scanner vendors, acquisition protocols, and patient demographics. This setting represents a clinically meaningful cross-center domain shift, where models must adapt to systematic variations across institutions without data sharing. We use this dataset to assess the robustness of our approach under realistic inter-center heterogeneity and high-dimensional 3D imaging conditions.

The **OfficeHome** dataset [31] is a widely used benchmark for domain adaptation; it provides a standardized and well-recognized testbed for evaluating the generalization and scalability of our method beyond the clinical domain, demonstrating that the proposed mechanism remains effective in conventional DIL tasks.

Table 3. Details of multi-domain datasets used in our study.

Dataset	Domain	Train	Test
Skin Pathology Diagnosis	PH2	160	40
	MSK	2322	581
	D7P	769	193
	HAM	8012	2003
	BCN	9930	2483
	Others	4952	1239
	Dermoscopic	4232	1058
Cyst-X	NYU	119	31
	MCF	106	28
	NU	147	39
	AHN	12	4
	MCA	19	5
	IU	60	14
	EMC	62	16
OfficeHome	Art	1942	485
	Clipart	3492	873
	Product	3551	888
	Real-World	3486	871

10.2. Implementation Details

All experiments were conducted using the same training configuration unless otherwise specified. Each domain was trained for **20 epochs** with a batch size of **64**. The optimization employed the **AdamW** optimizer with an initial learning rate of **0.03** and a **cosine annealing** scheduler. All experiments were performed on an **NVIDIA H800** GPU. These consistent settings ensure that performance differences across domains and datasets arise solely from model behavior rather than training hyperparameters.

Across all datasets, we perform stratified sampling with a **4:1** ratio to split each domain into training and testing sets, ensuring consistent class distributions across both stages. When multiple domains belong to the same dataset, we apply identical preprocessing and augmentation strategies to guarantee fairness in comparison. The domain names and the corresponding numbers of training and testing samples for each dataset are summarized in Table 3. All images are resized to 224×224 for 2D inputs or a fixed 3D resolution for Cyst-X before being fed into the model.

10.3. Evaluation Metrics

For each dataset with N_t domains, training proceeds sequentially. After completing the training on domain t, we evaluate the model on all previously seen domains $\{1, \ldots, t-1, t\}$, yielding an accuracy sequence $[a_1, a_2, ..., a_{N_t}]$. For a given random seed s, we compute the per-seed average accuracy $\overline{A}^{(s)}$ and final accuracy $A_T^{(s)}$ from this sequence:

$$\overline{A}^{(s)} = \frac{1}{N_t} \sum_{t=1}^{N_t} a_t^{(s)}, \quad A_T^{(s)} = a_{N_t}^{(s)} \tag{25}$$

To ensure reliability, all experiments are repeated using five random seeds. In Table 1, we report the mean and standard deviation of $\overline{A}^{(s)}$ and $A_T^{(s)}$ across the five seeds, denoted as $\overline{A}_{\pm\text{std}}$ and $A_{T\,\pm\text{std}}$, respectively, to provide statistically robust comparisons. In addition, we measure the ratio of trainable parameters to the total number of parameters to quantify the parameter efficiency of each method. To assess the statistical significance of performance differences, we compare $\overline{A}$ and A_T of our method against each baseline using paired t-tests and report the resulting p-values.

10.4. Extension to 3D MRI Data

To adapt the proposed framework to 3D volumetric data (Cyst-X benchmark), we replace the standard 2D patch embedding layer in the Vision Transformer backbone with a 3D patch embedding block implemented based on the *PatchEmbeddingBlock* from the MONAI library [4]. This design performs volumetric tokenization along three spatial dimensions while preserving the transformer encoder architecture, attention computation, and all downstream components unchanged. Such a modification allows our framework to handle 3D medical images in a manner consistent with its 2D formulation, maintaining both representational coherence and parameter efficiency.

10.5. Clarification on Replay Settings

We emphasize that, except for the DGR baseline, all compared approaches are evaluated under a strictly **rehearsal-free** setting, i.e., no data samples from past domains are stored or replayed. This setting reflects the practical constraint of non-revisitable data in dynamic medical imaging environments and aligns with the dynamic DIL protocol defined in the main paper.

Algorithm 1 The Procedure of DK-DDIL

Require: Sequential domains $\{\mathcal{D}_t\}_{t=1}^{N_t}$ with label sets $\{\mathcal{Y}_t\}_{t=1}^{N_t}$; pretrained backbone f_θ; maximum and minimum ranks $r_{\max}, r_{\min}$.

Ensure: DAMs $\{\mathcal{A}^{(i)}\}_{i=1}^{N_t}$ and accumulated prototype memory $\mathbf{P}^{(N_t)}$.

Initialize prototype memory $\mathbf{P}^{(0)} \leftarrow \emptyset$ and DAM $\mathcal{A}^{(1)}$ with rank $r_{\max}$.

for $t = 1$ **to** N_t **do** ▷ Adapt to new domain $\mathcal{D}_t$

Init classifier head for new label set $\mathcal{Y}_t$

MKI:

Compute cosine-annealed coefficient: $\alpha_e = \alpha_{\text{final}} + (\alpha_{\text{init}} - \alpha_{\text{final}}) \frac{1+\cos(\pi e/E)}{2}$

Fuse transferable low-rank parameters:

$$B^{(t)} \leftarrow \alpha_e \, B^{(t)} + \frac{(1-\alpha_e)}{t-1} \sum_{k=1}^{t-1} B^{(k)},$$

Keep other parameters $\{A^{(t)}\}$ unchanged for domain-specific adaptation.

DAM:

for each mini-batch $(\mathbf{x}_i^{(t)}, y_i^{(t)}) \in \mathcal{D}_t$ **do**

Extract feature $\mathbf{f}_i = f_\theta(\mathbf{x}_i^{(t)})$

Compute rank scores $\mathbf{s} \in \mathbb{R}^{r_{\max}}$ and mask $\mathbf{m} = \mathrm{STE}(\sigma(\mathbf{s}), \tau)$

Ensure $\sum_i m_i \geq r_{\min}$ by activating top-$r_{\min}$ entries

Apply masked low-rank update $\Delta W_m = A \, \mathrm{diag}(\mathbf{m}) \, B$

Forward pass with adaptive scaling $h = Wx + \alpha_t \cdot \Delta W_m x$, where $\alpha_t = \frac{r_{\max}}{\sum_i m_i}$

Compute sparsity regularization $\mathcal{L}_{\text{reg}} = \lambda_{\text{reg}} \frac{1}{r_{\max}} \sum_i \sigma(s_i)$

Update $\{A, B, \mathbf{s}\}$ by minimizing $\mathcal{L}_{\text{reg}}$

end for

DCL:

Compute normalized features $\mathbf{f}_i$ and prototypes $\mathbf{P}^{(t)}$

Calculate: $\mathcal{L}_{\text{pos}}$, $\mathcal{L}_{\text{neg-intra}}$, $\mathcal{L}_{\text{neg-cross}}$, $\mathcal{L}_{\text{intra}}$

Combine into

$$\mathcal{L}_{\text{DCL}} = \mathcal{L}_{\text{pos}} + \frac{s}{S_t}\big(\mathcal{L}_{\text{neg-intra}} + \mathcal{L}_{\text{neg-cross}}\big) + \mathcal{L}_{\text{intra}}$$

Update $\{A, B, \mathbf{s}\}$ by minimizing $\mathcal{L}_{\text{DCL}}$

Update class prototypes $\mathbf{p}_c^{(t)}$ using Eq. (9)

Append to memory $\mathbf{P}^{(t)} = \mathbf{P}^{(t-1)} \cup \{\mathbf{p}_c^{(t)}\}$

Freeze current adapter $\mathcal{A}^{(t)}$ and store best checkpoint

end for

return DAMs $\{\mathcal{A}^{(i)}\}_{i=1}^{N_t}$ and accumulated prototypes $\mathbf{P}^{(N_t)}$