

Aligning Prototypes and Updating Null Spaces for Continual WSI Learning

Xianrui Li¹, Kaiwen Xiao², and Antoni B. Chan¹(✉)

¹ Department of Computer Science, City University of Hong Kong, Hong Kong
xianrui.li@my.cityu.edu.hk, abchan@cityu.edu.hk

² LBP SingPath AI Lab, Singapore

Abstract. Whole Slide Image (WSI) analysis in digital pathology requires continual learning (CL) to adapt to growing data streams without catastrophic forgetting. Current CL methods for WSIs face challenges from large sample sizes and strict patient privacy constraints, risking either data leakage or limited performance on visually similar categories. Existing methods suffer from inter-task class confusion due to insufficient discriminative supervision, and an inability to update prior class representations. To address these, we propose AlignMIL-NS with two innovations: (1) Prototype-Class Feature Alignment (PCFA), which enforces within-task alignment between class and visual features while promoting inter-task discrimination; and (2) Null Space Class Feature Update (NS-CFU), which refines previous task representations within their task-specific null spaces to mitigate interference from new tasks. On a challenging five-dataset WSI benchmark, AlignMIL-NS achieves 89.1% AACC on both task orders, outperforming all rehearsal-free methods and matching rehearsal-based approaches without storing any patient data.

Keywords: Continual Learning · Multiple Instance Learning · Whole Slide Image

1 Introduction

Whole Slide Image (WSI) analysis is fundamental to digital pathology, enabling automated disease diagnosis, prognosis, and treatment response prediction. Due to the gigapixel nature of WSIs and the absence of instance-level annotations, Multiple Instance Learning (MIL) [8, 14, 19, 25, 5] has become the predominant paradigm, treating each slide as a bag of patches supervised only by slide labels. Recent integration of vision-language models (VLMs) into MIL frameworks [17, 24, 13] further enhances semantic alignment in pathology classification.

Despite their strong performance, existing MIL methods are designed for fixed data distributions and struggle when new disease subtypes or data modalities emerge over time. This motivates continual learning (CL) [26], which aims to sequentially acquire new knowledge without forgetting previously learned tasks. WSI-based CL faces particularly acute challenges: the number of available slides per task is small yet each slide is extremely large, and stringent patient privacy regulations make it infeasible to retain raw data across tasks [7, 10].

Existing CL strategies include regularization-based [9, 11], architecture-based [18], rehearsal-based [2, 12], and prompt/prototype-based methods [22, 21, 23]. Recent WSI-specific works (ConSlide [7], AKD-PMP [10], MICIL [15], and QPMIL-VL [6]) achieve competitive performance while respecting privacy constraints. However, applying general CL to WSI-based MIL remains challenging: instance-level labels are absent, bag sizes are large, and privacy constraints preclude most rehearsal techniques. Our analysis of prototype-based methods (e.g., QPMIL-VL) reveals two limitations: **(i)** insufficient discriminative supervision causes confusion when new classes are visually similar to prior ones; **(ii)** frozen old-task prototypes cannot adapt to interference from new tasks, degrading performance on challenging benchmarks.

To address these issues, we propose **AlignMIL-NS**, a prototype-based continual MIL framework incorporating two targeted improvements. First, **Prototype-Class Feature Alignment (PCFA)** enforces within-task alignment between learned prototypes and class-level features while simultaneously promoting cross-task discriminability, reducing confusion among visually similar categories across incremental steps. Second, **Null Space Class Feature Update** enables continuous refinement of previous-task class representations by constraining updates to the null space of each task’s original feature subspace, effectively suppressing interference from new tasks without corrupting previously acquired knowledge, representing a fundamentally different philosophy from conventional methods that freeze old representations to avoid forgetting. The primary contributions of this paper are summarized as follows:

- PCFA improves within-task alignment and cross-task separability for confusable classes.
- NS-CFU updates previous-task representations in null spaces to reduce interference.
- A five-dataset class-incremental benchmark with cross-task visual similarity.

2 Preliminaries

2.1 Class-incremental MIL Formulation

In the class-incremental MIL, the model sequentially learns a series of task sessions with MIL datasets $\mathcal{D} = \{\mathcal{D}_1, \mathcal{D}_2, \dots, \mathcal{D}_T\}$, where the t -th task’s dataset is $\mathcal{D}_t = \{(B_i^t, y_i^t, t)\}_{i=1}^{n_t}$. B_i^t is a bag, and $y_i^t \in C_t$ is the corresponding label from the set C_t .

2.2 Prototype-based MIL

Normally, a WSI bag is segmented into n patches, and the features of these patches are extracted by a vision encoder $Z = \{h_i\} = E_v(x_i) \in \mathbb{R}^{n \times D}$, where x_i represents the i -th patch. TOP [17] introduces a prototype-based pooling strategy to aggregate the instance features Z into a bag-level representation $f_b \in \mathbb{R}^D$.

$$f_b = \text{mean}(\text{Softmax}(Z \cdot f_p^\top)^\top \cdot Z), \quad (1)$$

where $f_p = E_l(P) \in \mathbb{R}^{N \times D}$ is the prototype features, encoded by a fixed language encoder from a set of learnable prompt templates $P \in \mathbb{R}^{N \times L}$. For conventional classification task, prototype-based MIL can be optimized via loss function

$$\mathcal{L}_C = -\log \frac{\exp[\tau \cdot \langle f_b, [f_c]_y \rangle]}{\sum_{k=1}^C \exp[\tau \cdot \langle f_b, [f_c]_k \rangle]}, \quad (2)$$

where C is the total number of classes, $[f_c]_k$ is the learnable k -th class feature, τ is a temperature parameter, and y is the true class.

However, existing prototype-based CL methods are incompatible with MIL: OnPro [23] requires instance-level labels, while L2P-type methods [22] cannot scale to tens of thousands of instances. QPMIL-VL [6] addresses this gap with a prototype pool $\{(k_i, f_{p_i})\}_{i=1}^M$ and prototype querying $f_q = \text{Pooling}(Z) \in \mathbb{R}^D$ to filter out prototypes suitable for each task, while also leveraging VLMs to enhance semantic expression in terms of labels $f_c \in \mathbb{R}^D$:

$$f_c = \text{mean}(E_l(T_c)) + \beta \mathbf{v}_c, \quad (3)$$

where $T_c \in \mathbb{R}^{m \times L}$ represents the text tokens corresponding to class descriptions, β is a multiplier and $\mathbf{v}_c \in \mathbb{R}^D$ is a learnable vector. Besides using the standard $\mathcal{L}_C$, optimization of the prototype-based MIL network in the class-incremental learning setting uses a prototype matching loss $\mathcal{L}_M$, which enhances relevance between query vectors and selected prototypes:

$$\mathcal{L}_M = -\frac{1}{N} \sum_{i=1}^N \langle f_q, k_i^t \rangle, \quad (4)$$

where k_i^t is the selected keys at task t .

3 Proposed Method

3.1 Analysis of Prototype-based MIL

Existing prototype-based MIL methods such as QPMIL-VL [6] suffer performance degradation when distinguishing visually similar classes across sequential tasks. Fig. 1(left) visualizes the average class-bag feature similarity over three tasks, revealing two critical issues: (1) current-task class features become excessively similar to bag features of previously learned tasks, sometimes surpassing the similarity to their own class (red circle, T3); and (2) frozen class features from previous tasks grow similar to bag features from subsequent tasks, causing symmetric confusion. Both issues stem from the lack of explicit cross-task discriminability constraints and the inability to update previous class representations, causing certain categories of catastrophic forgetting.

3.2 Prototype-Class Feature Alignment (PCFA)

QPMIL-VL lacks explicit constraints enforcing the alignment between current class features and visual features *while discriminating them from previously*

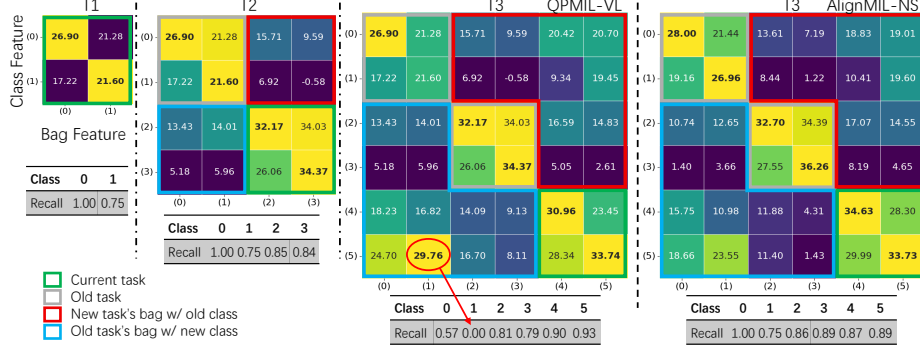


Fig. 1: Average feature similarity heatmaps of QPMIL-VL (left, T1-T3) and AlignMIL-NS (right, T3) across sequential task sessions (separated by black dashed lines). The x-axis and y-axis correspond to bag feature labels and class feature labels, respectively. Green/gray outlines indicate current/previous task sub-matrices; red outlines (upper off-diagonal) show new-task features misclassified as previous classes; blue outlines (lower off-diagonal) show previous-task features misclassified as new classes. Per-class recall is shown at the bottom.

learned visual features. Since historical visual features are inaccessible in the CL setting, we approximate them using prototype features f_p . Our proposed PCFA enforces alignment between current-task class features f_c^t and their corresponding prototype features f_p^t , while simultaneously discriminating against prototypes from previous tasks.

Formally, given a class feature f_c^t from task t , matched prototype features $\{f_{p_i}^t\}_{i=1}^N$, and negative prototype features from previous tasks $\{f_p^j\}_{j<t}$, the proposed PCFA loss is:

$$\begin{aligned}
 s_i^+ &= \langle f_c^t, f_{p_i}^t \rangle / \tau, \quad i \in \{1, \dots, N\}, \\
 s_k^- &= \langle f_c^t, f_{p_k}^j \rangle / \tau, \quad j < t, \\
 \mathcal{L}_{\text{PCFA}}(t) &= -\log \frac{\sum_{i=1}^N \exp(s_i^+)}{\sum_{i=1}^N \exp(s_i^+) + \sum_{j<t} \sum_k \exp(s_k^-)}. \quad (5)
 \end{aligned}$$

This loss maximizes intra-task feature alignment and enhances inter-task discrimination, effectively addressing catastrophic forgetting among visually similar classes, as shown in Fig. 2 (b). Compared with the InfoNCE formulation [16], our PCFA loss can handle multiple positive samples.

3.3 Null Space Class Feature Update

In existing frameworks, class features from previous tasks typically remain static during the training of subsequent tasks to prevent interference. However, this approach may yield suboptimal class representations due to early convergence, i.e., *they cannot take into account the changes of the optimal discriminative class*

boundaries due to the introduction of new classes. To refine these previous-task class representations to be dissimilar to new-task class features without impairing the previous task, we propose updating class features exclusively within their task-specific null spaces after training each task.

The null space ϕ_j for class j is defined as the directions orthogonal to the visual features for class j , i.e., the null space of the query feature matrix F_q^j aggregated from all WSIs in class j . Any change of the class feature f_c^j in directions of ϕ_j will minimally change its response to visual features from the j -th class, thus preserving the j -th task’s knowledge. Fig. 2 (c) illustrates this update process. Specifically, for each previously learned task j , we enforce updating the class feature f_c^j along its null space ϕ_j to satisfy:

$$\langle f_q^i, f_c^j + \phi_j \alpha_j^\top \rangle < \langle f_q^i, f_c^j \rangle - \gamma, \quad \forall j < t, i \neq j, \quad (6)$$

where α_j is an optimizable update step, and γ is a margin hyperparameter. Note that the similarity within the j -th task is unchanged, as the null-space constraint implies: $\langle f_q^j, f_c^j + \phi_j \alpha_j^\top \rangle = \langle f_q^j, f_c^j \rangle$. The corresponding optimization loss for calculating α_j is:

$$\mathcal{L}_{\alpha_j} = \sum_{i \neq j} \text{ReLU}(\langle f_q^i, \phi_j \alpha_j^\top \rangle + \gamma). \quad (7)$$

After computing α_j , the class feature is then updated,

$$f_c^j \leftarrow f_c^j + \phi_j \alpha_j^\top. \quad (8)$$

Practically, an exact null space may not exist. We compute the task query matrix F_q by concatenating all query vectors from $\mathcal{D}_t$, then perform SVD: $(U, \Lambda, V^\top) = \text{SVD}(F_q)$. If $\text{rank}(F_q) < \min(\text{rows}(F_q), \text{cols}(F_q))$, the null space ϕ_t is spanned by the columns of V corresponding to zero singular values; otherwise, we approximate it using the subspace of the ten smallest singular values following [20]. We store ϕ_t for subsequent sessions. The learnable parameter α is then optimized via the loss in Eq. 7.

3.4 Overall Framework

As depicted in Fig. 2(a), the final loss is defined as:

$$\mathcal{L} = \mathcal{L}_C + \lambda_M \mathcal{L}_M + \lambda_{\text{PCFA}} \mathcal{L}_{\text{PCFA}}. \quad (9)$$

During training on task t , each WSI is processed through prototype querying, aggregation (Eq. 1), and classification (Eq. 2), with PCFA applied per batch; previous class features are simultaneously refined in their respective null spaces via Eq. 8. After each task session concludes, the null space for the current task is computed via SVD for use in subsequent sessions.

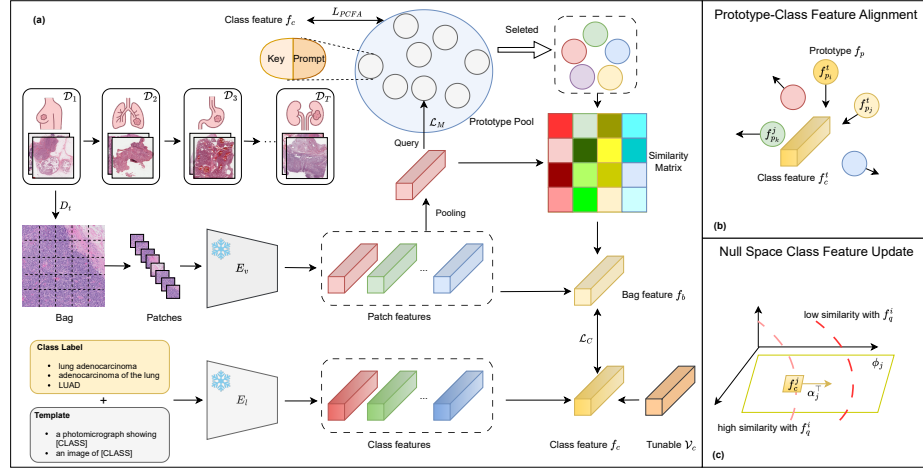


Fig. 2: Overview of AlignMIL-NS. (a) Pipeline: patch feature extraction (E_v), prototype querying and aggregation, and class feature generation (E_l , tunable v_c). (b) Prototype-class feature alignment. (c) Null space class feature update.

4 Experiments

4.1 Experiment Setup

New Class-Incremental Learning Benchmark. We introduce a challenging benchmark of five WSI datasets: Camelyon16 [1], TCGA-LUNG, TCGA-BRCA, TCGA-ESCA, and TCGA-RCC [3]. Camelyon and BRCA share breast cancer origin but differ in tissue type, posing inter-task similarity challenges; RCC spans three subtypes, adding intra-task complexity. Performance is assessed under two orderings: **Forward** (Camelyon $\rightarrow$ LUNG $\rightarrow$ BRCA $\rightarrow$ ESCA $\rightarrow$ RCC) and **Reverse** (same sequence inverted). Patches (256×256 at $20\times$ magnification) are extracted via CLAM [14] and encoded into 512-d features by the visual encoder of CONCH [13], whose language encoder is used to generate class features (Eq. 3). Evaluation is conducted using patient-level 5-fold cross-validation (4 folds: 80% train / 20% val; 1 fold for testing); statistics are in Tab. 2.

Implementation Details. All models are trained with Adam (lr=1e-3) for 12 epochs per session, with $N = 5$ prototypes assigned per session (25 total). For AlignMIL-NS, we set memory pool size $M = 25$, null space threshold $\gamma = 0.4$, and loss weights $\lambda_M = 0.5$, $\lambda_{PCFA} = 1.0$. Tab. 4 shows that AlignMIL-NS reduces parameters by $15\times$ over QPMIL-VL by replacing the language-encoded prototypes $f_p = E_l(P)$ with directly learnable vectors $f_p \in \mathbb{R}^{N \times D}$, while maintaining identical FLOPs (11-class inference, 10k patches/slide).

Evaluation Metrics. We report three metrics after all T tasks. Average accuracy $\mathbf{AACC} = \frac{1}{T} \sum_{j=1}^T a_{T,j}$ measures final accuracy. Backwards transfer $\mathbf{BWT} = \frac{1}{T-1} \sum_{j=1}^{T-1} (a_{T,j} - a_{j,j})$ quantifies forgetting. Intransience measure $\mathbf{IM}$

Table 1: Continual learning results (%). AACC: mean $\pm$ std over 5 folds; BWT and IM: mean only. Methods marked @ N use a replay buffer of N slides.

Method	Forward-order			Reverse-order		
	AACC $\uparrow$	BWT $\uparrow$	IM $\downarrow$	AACC $\uparrow$	BWT $\uparrow$	IM $\downarrow$
Joint Train (upper bound)	90.3 $\pm$ 1.6	—	—	90.3 $\pm$ 1.6	—	—
Fine-tune (lower bound)	19.6 $\pm$ 1.1	-89.4	-0.8	20.0 $\pm$ 0.7	-88.7	-0.7
EWC	19.8 $\pm$ 1.2	-88.9	-0.7	20.1 $\pm$ 0.8	-88.5	-0.7
LwF	20.1 $\pm$ 1.4	-88.0	-0.5	21.3 $\pm$ 1.4	-87.4	-0.3
MICIL	23.1 $\pm$ 2.4	-88.3	-0.1	19.0 $\pm$ 0.5	-92.9	0.3
ER@30	71.6 $\pm$ 5.1	-28.4	-0.8	71.9 $\pm$ 4.7	-28.6	-1.2
DER++@30	73.1 $\pm$ 3.6	-27.1	-1.2	73.4 $\pm$ 6.6	-27.5	-1.8
ConSlide@30	76.4 $\pm$ 1.0	-21.9	5.1	86.0 $\pm$ 1.3	-8.4	-1.8
PMP-AKD@30	90.0 $\pm$ 1.4	-5.5	-4.1	89.1 $\pm$ 2.4	-7.1	-4.3
QPMIL-VL	79.1 $\pm$ 5.3	-15.1	-0.7	72.7 $\pm$ 2.2	-22.8	-0.7
AlignMIL-NS (ours)	89.1 $\pm$ 1.4	-4.0	-1.8	89.1 $\pm$ 1.4	-3.6	-0.9

$= \frac{1}{T} \sum_{j=1}^T (a_j^* - a_{j,j})$ measures plasticity relative to joint training, where $a_{T,j}$ is accuracy on task j after training all tasks, $a_{j,j}$ is accuracy right after learning task j , and a_j^* is joint-training accuracy.

Baseline Methods. Fine-tuning (AlignMIL-NS w/o $\mathcal{L}_M$) and joint training serve as lower/upper bounds (*Baseline*). *Regularization* methods include EWC [9], LwF [11], and MICIL [15]. *Rehearsal-based* methods include ER [4], DER++ [2], ConSlide [7], and AKD-PMP [10]. QPMIL-VL [6] is the *Prototype-based* baseline most related to our work.

4.2 Results

Rehearsal-Free Superiority. AlignMIL-NS achieves AACC of 89.1% on both task orders, substantially outperforming rehearsal-free baselines (EWC: 19.8%/20.1%, LwF: 20.1%/21.3%, MICIL: 23.1%/19.0%) and QPMIL-VL (79.1%/72.7%), with BWT of -4.0/-3.6 (Fwd/Rev), far closer to zero than QPMIL-VL (-15.1/-22.8), confirming both strong plasticity and reduced forgetting.

Robustness and Rehearsal Parity. Unlike QPMIL-VL (79.1% $\rightarrow$ 72.7% under order reversal), AlignMIL-NS maintains 89.1% on both orderings. It also matches PMP-AKD (90.0%/89.1%, buffer=30 slides) without storing any patient data, eliminating privacy and storage concerns.

4.3 Ablation Studies

Tab. 3 ablates each component on the forward-order benchmark.

Baseline Performance. Without both components (AACC 77.3%), the language encoder from QPMIL-VL can be safely removed without degradation.

Impact of Null Space Class Feature Update. Adding NS-CFU alone (“w/o PCFA”) raises AACC to 83.6%, confirming that null-space updates refine

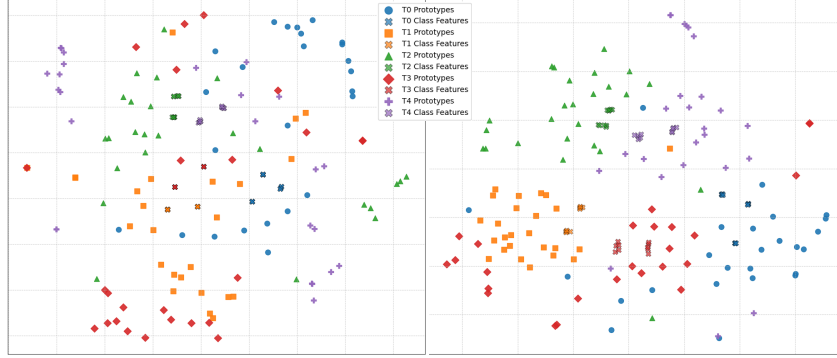


Fig. 3: Prototype and class feature visualization of QPMIL-VL (left) and our AlignMIL-NS (right). In the same task, prototype and class features are represented by the same color. The class features are represented as $\times$'s.

classifier boundaries. This gain is robust to the margin hyperparameter γ : NS-CFU consistently improves over the baseline for $\gamma \in [0, 0.6]$.

Impact of PCFA. PCFA alone ("w/o Null Space") yields AACC 87.9% (+10.6%), highlighting the critical role of cross-task feature alignment.

Table 2: Dataset statistics.

Dataset	Label	Slides
Camelyon16	Normal	239
	Tumor	160
LUNG	LUAD	493
	LUSC	484
BRCA	IDC	761
	ILC	191
ESCA	ESAD	84
	ESCC	66
RCC	CCRCC	519
	PRCC	297
	CHRC	121

Table 3: Ablation Study (AACC, %).

Configuration	AACC
w/o PCFA & Null Space	77.3 ± 5.1
w/o PCFA	83.6 ± 5.2
w/o Null Space	87.9 ± 1.7
AlignMIL-NS (full model)	89.1 ± 1.4

Table 4: Model parameters and FLOPs comparison.

Model	Params (M)	FLOPs (M)
QPMIL-VL	0.479	57.62
AlignMIL-NS	0.031	57.62

4.4 Feature Visualization

Fig. 3 shows t-SNE visualizations (reverse-order, 5-fold). AlignMIL-NS forms tighter prototype clusters and class features closely align with their prototypes. In contrast, QPMIL-VL’s class features in Tasks 3 and 4 deviate from their prototypes, reflecting weaker alignment and reduced discriminative power.

5 Conclusion

We presented AlignMIL-NS, a privacy-preserving continual MIL framework that combines PCFA and NS-CFU to mitigate catastrophic forgetting in WSI analysis, demonstrating strong knowledge retention and adaptability on a challenging benchmark. A current limitation is that the joint-training upper bound of AlignMIL-NS still trails static MIL models such as CLAM [14] and HIT [7]; closing this gap is our primary direction for future work.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Bejnordi, B.E., Veta, M., Diest, P.J.v., Ginneken, B.v., Karssemeijer, N., Litjens, G., Laak, J.A.v.d., Hermesen, M., Manson, Q.F., Balkenhol, M., Geessink, O.G.F., Stathonikos, N., Dijk, M.V.v., Bult, P., Beca, F., Beck, A.H., Wang, D., Khosla, A., Gargeya, R., Irshad, H., Zhong, A., Dou, Q., Li, Q., Chen, H., Lin, H., Heng, P., Hass, C., Bruni, E., Wong, Q., Halici, U., Öner, M.Ü., Cetin-Atalay, R., Berseth, M., Khvatkov, V., Vylegzhanin, A., Kraus, O.Z., Shaban, M., Rajpoot, N., Awan, R., Sirinukunwattana, K., Qaiser, T., Tsang, Y., Tellez, D., Annuscheit, J., Hufnagel, P., Valkonen, M., Kartasalo, K., Latonen, L., Ruusuvuori, P., Liimatainen, K., Albarqouni, S., Mungal, B., George, A., Demirci, S., Navab, N., Watanabe, S., Seno, S., Takenaka, Y., Matsuda, H., Phoulady, H.A., Kovalev, V., Kalinovskiy, A., Liauchuk, V., Bueno, G., Fernández-Carrobles, M., Serrano, I., Deniz, O., Racoceanu, D., Venâncio, R.: Diagnostic Assessment of Deep Learning Algorithms for Detection of Lymph Node Metastases in Women With Breast Cancer. *JAMA* **318**, 2199–2210 (2017)
2. Buzzega, P., Boschini, M., Porrello, A., Abati, D., Calderara, S.: Dark experience for general continual learning: a strong, simple baseline. In: *Conference on Neural Information Processing Systems* (2020)
3. Cancer Genome Atlas Research Network, J., et al.: The cancer genome atlas pan-cancer analysis project. *Nat. Genet* **45**(10), 1113–1120 (2013)
4. Chaudhry, A., Rohrbach, M., Elhoseiny, M., Ajanthan, T., Dokania, P.K., Torr, P.H., Ranzato, M.: On tiny episodic memories in continual learning. *arXiv preprint arXiv:1902.10486* (2019)
5. Cui, Y., Liu, Z., Liu, X., Liu, X., Wang, C., Kuo, T.W., Xue, C.J., Chan, A.B.: Bayes-MIL: A New Probabilistic Perspective on Attention-based Multiple Instance Learning for Whole Slide Images. In: *International Conference on Learning Representations* (2023)
6. Gou, J., Ji, L., Liu, P., Ye, M.: Queryable Prototype Multiple Instance Learning with Vision-Language Models for Incremental Whole Slide Image Classification. *Proceedings of the AAAI Conference on Artificial Intelligence* **39**(3), 3158–3166 (2025)
7. Huang, Y., Zhao, W., Wang, S., Fu, Y., Jiang, Y., Yu, L.: Conslide: asynchronous hierarchical interaction transformer with breakup-reorganize rehearsal for continual whole slide image analysis. In: *2023 IEEE/CVF International Conference on Computer Vision*. pp. 21292–21303. IEEE (2023)

8. Ilse, M., Tomczak, J.M., Welling, M.: Attention-based Deep Multiple Instance Learning. In: International Conference on Machine Learning. pp. 2132–2141 (2018)
9. Kirkpatrick, J., Pascanu, R., Rabinowitz, N., Veness, J., Desjardins, G., Rusu, A.A., Milan, K., Quan, J., Ramalho, T., Grabska-Barwinska, A., Hassabis, D., Clopath, C., Kumaran, D., Hadsell, R.: Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences* **114**(13), 3521–3526 (2017)
10. Li, X., Cui, Y., Li, J., Chan, A.B.: Advancing multiple instance learning with continual learning for whole slide imaging. In: *Proceedings of the Computer Vision and Pattern Recognition Conference*. pp. 20800–20809 (2025)
11. Li, Z., Hoiem, D.: Learning without Forgetting. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **40**(12), 2935–2947 (2018)
12. Lopez-Paz, D., Ranzato, M.: Gradient episodic memory for continual learning. *Advances in neural information processing systems* **30** (2017)
13. Lu, M.Y., Chen, B., Williamson, D.F.K., Chen, R.J., Liang, I., Ding, T., Jaume, G., Odintsov, I., Le, L.P., Gerber, G., Parwani, A.V., Zhang, A., Mahmood, F.: A visual-language foundation model for computational pathology. *Nature Medicine* **30**(3), 863–874 (2024)
14. Lu, M.Y., Williamson, D.F.K., Chen, T.Y., Chen, R.J., Barbieri, M., Mahmood, F.: Data-efficient and weakly supervised computational pathology on whole-slide images. *Nature Biomedical Engineering* **5**(6), 555–570 (2021)
15. Meseguer, P., Amor, R.D., Naranjo, V.: Micil: Multiple-Instance Class-Incremental Learning for skin cancer whole slide images. *Artificial Intelligence in Medicine* **152**, 102870 (2024)
16. Oord, A.v.d., Li, Y., Vinyals, O.: Representation Learning with Contrastive Predictive Coding. *arXiv.org abs/1807.03748* (2018)
17. Qu, L., Luo, X., Fu, K., Wang, M., Song, Z.: The Rise of AI Language Pathologists: Exploring Two-level Prompt Learning for Few-shot Weakly-supervised Whole Slide Image Classification. In: *Conference on Neural Information Processing Systems* (2023)
18. Rusu, A.A., Rabinowitz, N.C., Desjardins, G., Soyer, H., Kirkpatrick, J., Kavukcuoglu, K., Pascanu, R., Hadsell, R.: Progressive neural networks. *arXiv preprint arXiv: 1606.04671* (2016)
19. Shao, Z., Bian, H., Chen, Y., Wang, Y., Zhang, J., Ji, X., Zhang, Y.: Transmil: Transformer based Correlated Multiple Instance Learning for Whole Slide Image Classification. In: *Conference on Neural Information Processing Systems*. pp. 2136–2147 (2021)
20. Wang, S., Li, X., Sun, J., Xu, Z.: Training Networks in Null Space of Feature Covariance for Continual Learning. In: *Computer Vision and Pattern Recognition*. pp. 184–193 (2021)
21. Wang, Z., Zhang, Z., Ebrahimi, S., Sun, R., Zhang, H., Lee, C.Y., Ren, X., Su, G., Perot, V., Dy, J.G., Pfister, T.: Dualprompt: complementary prompting for rehearsal-free continual learning. *European Conference on Computer Vision* (2022)
22. Wang, Z., Zhang, Z., Lee, C.Y., Zhang, H., Sun, R., Ren, X., Su, G., Perot, V., Dy, J.G., Pfister, T.: Learning to Prompt for Continual Learning. In: *Computer Vision and Pattern Recognition*. pp. 139–149 (2022)
23. Wei, Y., Ye, J., Huang, Z., Zhang, J., Shan, H.: Online Prototype Learning for Online Continual Learning. In: *IEEE International Conference on Computer Vision*. pp. 18718–18728 (2023)

24. Xu, H., Usuyama, N., Bagga, J., Zhang, S., Rao, R., Naumann, T., Wong, C., Gero, Z., González, J., Gu, Y., Xu, Y., Wei, M., Wang, W., Ma, S., Wei, F., Yang, J., Li, C., Gao, J., Rosemon, J., Bower, T., Lee, S., Weerasinghe, R., Wright, B.J., Robicsek, A., Piening, B., Bifulco, C., Wang, S., Poon, H.: A whole-slide foundation model for digital pathology from real-world data. *Nature* **630**(8015), 181–188 (2024)
25. Zhang, H., Meng, Y., Zhao, Y., Qiao, Y., Yang, X., Coupland, S.E., Zheng, Y.: Dtf-d-MIL: Double-Tier Feature Distillation Multiple Instance Learning for Histopathology Whole Slide Image Classification. In: *Computer Vision and Pattern Recognition*. pp. 18780–18790 (2022)
26. Zhou, D.W., Sun, H.L., Ning, J., Ye, H.J., Zhan, D.C.: Continual Learning with Pre-Trained Models: A Survey. In: *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*. vol. abs/2401.16386. International Joint Conferences on Artificial Intelligence Organization (2024)