# Advancing Multiple Instance Learning with Continual Learning for Whole Slide Imaging

Xianrui Li
Dept. of Computer Science
City University of Hong Kong
xianrui.li@my.cityu.edu.hk

Yufei Cui
Noah's Ark Lab, Huawei Canada
Montreal, Quebec, Canada
yufei.cui@huawei.com

Jun Li
Guangzhou Bingli Technology Co., Ltd.
Guangzhou, Guangdong

Antoni B. Chan
Dept. of Computer Science
City University of Hong Kong
abchan@cityu.edu.hk

## Abstract

*Advances in medical imaging and deep learning have propelled progress in whole slide image (WSI) analysis, with multiple instance learning (MIL) showing promise for efficient and accurate diagnostics. However, conventional MIL models often lack adaptability to evolving datasets, as they rely on static training that cannot incorporate new information without extensive retraining. Applying continual learning (CL) to MIL models is a possible solution, but often sees limited improvements. In this paper, we analyze CL in the context of attention MIL models and find that the model forgetting is mainly concentrated in the attention layers of the MIL model. Using the results of this analysis we propose two components for improving CL on MIL: Attention Knowledge Distillation (AKD) and the Pseudo-Bag Memory Pool (PMP). AKD mitigates catastrophic forgetting by focusing on retaining attention layer knowledge between learning sessions, while PMP reduces the memory footprint by selectively storing only the most informative patches, or "pseudo-bags" from WSIs. Experimental evaluations demonstrate that our method significantly improves both accuracy and memory efficiency on diverse WSI datasets, outperforming current state-of-the-art CL methods. This work provides a foundation for CL in large-scale, weakly annotated clinical datasets, paving the way for more adaptable and resilient diagnostic models.*

## 1. Introduction

The advancement of AI technology has significantly transformed medical imaging, with notable progress in whole slide image (WSI) analysis [28]. In particular, developments in multiple instance learning (MIL) have enhanced the diagnostic accuracy and efficiency of deep learning models applied to WSIs [15, 29], substantially reducing diagnostic time and minimizing misdiagnosis rates. Despite these advancements, most MIL methodologies currently rely on a static framework [5, 8], wherein models are optimized using a fixed dataset. This static approach creates a critical limitation in terms of knowledge updating. As medical technology evolves rapidly, new data is continually generated [3, 4, 13], often with variations in imaging equipment, acquisition methods, and standardized settings. These factors alter the labels and distribution patterns of new datasets, making it essential for models to adapt to these changes [21]. Full retraining of models using both new and historical data, however, is resource-intensive and time-consuming. Consequently, continual learning (CL) [7, 24] emerges as an efficient solution to this challenge, enabling models to incrementally integrate new information and thereby maintain their diagnostic relevance without the prohibitive costs of full retraining. However, CL faces catastrophic forgetting [11], where models lose previously acquired knowledge when learning new information, addressed by regularization-based, architectural-based, and replay-based methods [22].

Our research focuses on class-incremental learning (CIL), a specialized area within CL that holds particular promise for applications in medical imaging. As medical knowledge advances, disease annotations are becoming increasingly detailed, leading to a proliferation of diagnostic categories. This development creates a pressing need for models capable of continually incorporating new classes of medical information [5] while retaining previously acquired knowledge, i.e., class-incremental learning. Moreover, re-

search suggests that models trained across multiple medical domains often achieve superior performance compared to those restricted to single domains [6], underscoring the potential of multi-domain learning to enhance diagnostic accuracy and generalizability.

Applying CL to WSI presents unique challenges due to the substantial data complexity and annotation demands associated with WSIs. Each WSI contains gigabytes of pixel data, making detailed annotations costly and time-consuming due to required expert input. Most WSIs only have slide-level labels without detailed annotations. To address this limitation, MIL has become a widely adopted strategy in WSI analysis. In the MIL paradigm, each WSI is conceptualized as a "bag" of smaller "patches" allowing models to be trained using high-level, whole-slide labels rather than exhaustive pixel- or patch-level annotations. In particular, *attention* MIL [9, 15, 20, 30] first uses an attention network to obtain an attention weight on each patch, with which a slide-level feature vector is calculated via the weight and the patch-level features. Finally, the slide-level feature serves as the input to a slide-level classifier.

Recent advancements have begun to explore the integration CIL within the MIL framework, with encouraging results [5, 8]. Some studies have proposed new frameworks specifically designed for CIL in WSI, yet these often require substantially more parameters and show only moderate performance improvements in conventional settings. Other approaches attempt to adapt existing CIL techniques for use with MIL models, also yielding only limited enhancements. Through our investigation of MIL under CIL, we have identified a critical insight: the patterns of forgetting in attention MIL classification models under CIL differ significantly from those observed in traditional image classification models. Specifically, our theoretical analysis and model visualizations indicate that *model forgetting in MIL is concentrated primarily in the attention network*, rather than in other layers like the classifier (logit) layer.

Based on our findings, we propose two effective and adaptable methods that improve the performance of traditional MIL models in CIL scenarios. The first method, attention knowledge distillation (AKD), addresses forgetting by imposing constraints on the attention layers during CL training. By using distillation techniques to transfer knowledge from previous models to the current one, AKD balances the retention of prior knowledge with the assimilation of new information. The second method, the pseudo-bag memory pool (PMP), mitigates the high memory demands of replaying WSI data by using a patch importance scoring technique to selectively sample critical patches, thereby substantially reducing memory requirements. Together, AKD and PMP provide a robust framework for enhancing MIL in CL settings, addressing both knowledge retention and memory efficiency while accommodating the specific challenges posed by WSIs.

In summary, our work presents three major contributions to MIL and continual learning research:

- To the best of our knowledge, this is the first attempt to theoretically and experimentally explore the fundamental reasons for the catastrophic forgetting of attention MIL in CIL. Compared to classical classification models where forgetting occurs mainly in the classification layer, we show that attention MIL experiences forgetting primarily in the attention layer.
- We have developed a specialized knowledge distillation strategy tailored for attention networks in MIL, which significantly enhances model performance during CL, empirically approaching the upper limit achievable through joint training.
- We introduce a novel pseudo-bag memory pool method, which effectively facilitates the implementation of replay-based approaches in MIL, offering a new direction for memory management in this context.

## 2. Related Work

### 2.1. Multiple Instance Learning (MIL)

Many MIL models leverage bag representations to predict the slide label. For instance, ABMIL [9] generates a bag representation by calculating weighted patch features, enabling it to focus on the most informative patches within a slide. Building on this approach, CLAM [15] refines this with instance clustering for multi-category tasks. TransMIL [20] proposes a correlated MIL framework that integrates multi-head self-attention and spatial information to capture global correlations across the an image. Similarly, DTFD [29] employs a two-tier framework with ABMIL as the foundational module in each tier, aiming to capture hierarchical features across different levels of granularity.

While these models demonstrate strong performance on static datasets, they generally do not account for scenarios involving non-stationary data. In dynamic environments where data distributions may shift over time, traditional MIL approaches such as CLAM [15] and TransMIL [20] are vulnerable to issues like catastrophic forgetting.

### 2.2. Continual Learning (CL)

CL methodologies in computer vision are generally categorized into three main approaches. The first, regularization-based methods, seek to address catastrophic forgetting by constraining the gradients of model parameters like EWC [11] and MAS [1]. However, these methods often produce suboptimal results in complex environments like realistic pathology data. The second category, model architecture-based methods, involves modifying network structures through strategies like compression and expansion to incorporate new knowledge without overwriting ex-

isting information. Approaches such as PNN [19] and PackNet [16] exemplify this technique, which, while theoretically robust for retaining prior knowledge, tends to suffer from scalability issues as an extension of the MIL model is not as simple as the classical classification model. Finally, replay-based methods utilize selective reintroduction of past data samples to reinforce memory, helping models to maintain prior knowledge during the training of new tasks, like DER [2] and GEM [14]. Nevertheless, these methods face challenges related to memory pool configuration and sample selection, which are critical since WSI datasets only contain hundreds of samples, and a WSI may be too large to store in the memory.

### 2.3. Class-incremental learning in MIL

Recent studies have increasingly explored the application of class-incremental learning (CIL) within MIL. ConSlide [8] combines fast and slow learning, substituting the WSI buffer with a region buffer. This adaptation effectively reduces memory pool requirements and enables efficient data replay. Alternatively, MICIL [5] adopts a regularization-based strategy that builds on the LwF [12] paradigm.

In contrast to these approaches, through theoretical analysis and model visualizations, we identify that model forgetting is primarily concentrated in the *attention layers*. Based on our findings, we then propose two new methods that focus on preserving knowledge learned by the attention network, which provide a significant boost to MIL performance in CIL scenarios.

## 3. Motivation

### 3.1. Problem Statement

**MIL.** In the WSI MIL classification problem, the WSI $\mathcal{X}$ comprises a bag of patches, $\mathbf{P} = \{\mathbf{p}_n\}_{n=1}^N$, and is associated with the bag-label $y$, where $N$ is the number of patches, which varies with different bags. The objective of MIL is to predict the bag-level $y$ label with the patches in the bag.

We follow the standard formulation of ABMIL [9]. ABMIL classifies a WSI in three steps following the MIL paradigm. First, a WSI is divided into $N$ patches, and a bag of patch features is obtained via a fixed feature extractor $g(\cdot)$, $\mathbf{H} = \{\mathbf{h}_n\}_{n=1}^N$, where the $n$-th patch feature is $\mathbf{h}_n = g(\mathbf{p}_n) \in \mathbb{R}^d$, and $d$ is the dimension of the feature vector. Second, the patch features are aggregated to compute a bag-level (slide-level) feature $z$:

$$\mathbf{z} = \mathbf{H}\mathbf{a} = \sum_{n=1}^{N} a_n \mathbf{h}_n, \tag{1}$$

where $a_n$ is the normalized attention score of the $n$th patch,

$$a_n = \frac{\exp\left\{\mathbf{w}^\top(\tanh(\mathbf{V}_1\mathbf{h}_n) \circ \sigma(\mathbf{V}_2\mathbf{h}_n))\right\}}{\sum_{m=1}^{N}\exp\left\{\mathbf{w}^\top(\tanh(\mathbf{V}_1\mathbf{h}_m) \circ \sigma(\mathbf{V}_2\mathbf{h}_m))\right\}}, \tag{2}$$

Table 1. Test Accuracy of CLAM and TransMIL on Task 1 using different combinations of its attention/classifier from CL task sessions ($t$). The diagonal cells (blue) correspond to the accuracy of fine-tuned model $(\theta_t, \phi_t)$ after each task session.

| CLAM | | Attention $\theta_t$ | | |
|---|---|---|---|---|
| | | $t$=1 | $t$=2 | $t$=3 |
| Classifier $\phi_t$ | $t$=1 | 0.8621 | 0.1552 | 0.0517 |
| | $t$=2 | 0.8793 | 0.0000 | 0.0000 |
| | $t$=3 | 0.8448 | 0.0000 | 0.0000 |

| TransMIL | | Attention $\theta_t$ | | |
|---|---|---|---|---|
| | | $t$=1 | $t$=2 | $t$=3 |
| Classifier $\phi_t$ | $t$=1 | 0.8966 | 0.0345 | 0.0345 |
| | $t$=2 | 0.8966 | 0.0345 | 0.0000 |
| | $t$=3 | 0.8276 | 0.0345 | 0.0000 |

and $\theta = (\mathbf{w}, \mathbf{V}_1, \mathbf{V}_2)$ are learnable weight matrices, $\tanh$ and $\sigma$ refer to the hyperbolic tangent and sigmoid activation functions, and $\circ$ denotes the element-wise product. We denote $\mathbf{a} = [a_n]_n = f_\theta(\mathbf{H})$ as the attention vector, where $f_\theta$ is the attention network. Third, the bag label is predicted by a classifier $\hat{y} = f_\phi(\mathbf{z})$, where $f_\phi$ is an MLP.

**CIL.** CIL is a CL scenario consisting of training a model on tasks with disjoint label spaces. We define a sequence of datasets $\mathcal{D} = \{\mathcal{D}_1, \mathcal{D}_2, \cdots, \mathcal{D}_T\}$, where the $t$-th dataset $\mathcal{D}_t = \{\mathcal{X}, y, t\}_k$ contains tuples of the WSI sample $\mathcal{X}$, its corresponding label $y \in \mathcal{Y}_t$ (where $\mathcal{Y}_i \cap \mathcal{Y}_j = \emptyset$ for $i \neq j$), and the task identifier $t$ (available only during training). The subscript $k$ means there are $k$ samples for dataset $\mathcal{D}_t$. In CIL, the model needs to predict the classification label $y$ with WSI $\mathcal{X}$ as the input.

### 3.2. Decoupling Experiments

In this section, we investigate the cause of the performance degradation of CIL for MIL. Here we consider CLAM [15] and TransMIL [20] as a representative attention MIL. We define the attention network $\theta$ as the sub-network that computes the attention scores from the patch features Eq. (2) for CLAM and Xiong et al. [26] for TransMIL), and we define the classifier $\phi$ as the sub-network that takes the bag-level features $\mathbf{z}$ and predicts the class label $y$.

In our CL experiment after the $T$-th task session, we obtain a set of $T$ fine-tuned model $\{(\theta_t, \phi_t)\}_{t=1}^T$, which represents the evolution of the model when sequentially fine-tuning with new task data. To investigate the effect of CL, we then recombined the trained attention network $\theta_t$ and classifier $\phi_{t'}$ from different CL task sessions, and tested these "hybrid" models on Task 1 test data.

We use Camelyon-TCGA for the benchmark, as described in the Sec. 5.1. The results for all pairs of attention/classifier are presented in Tab. 1. The significant decline of Task 1 accuracy after training on subsequent tasks is evident in the diagonal cells of the tables. However, exam-

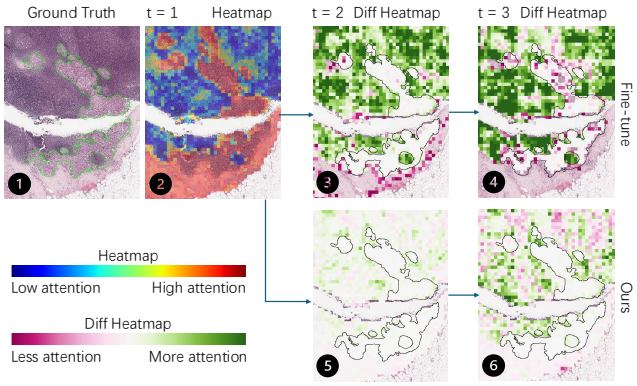


Figure 1. Example of the attention shift during CL for a Task 1 test image. (1) Input image with tumor area outlined in green. (2) The attention heat map for $t$=1 session. (3) and (4): the attention drift, relative to (2), after CL sessions $t \in \{2, 3\}$ for fine-tuning CL. The tumor areas are outlined in black. (5) and (6): the attention drift for our proposed method. Our method better preserves the attention much better than fine-tuning.

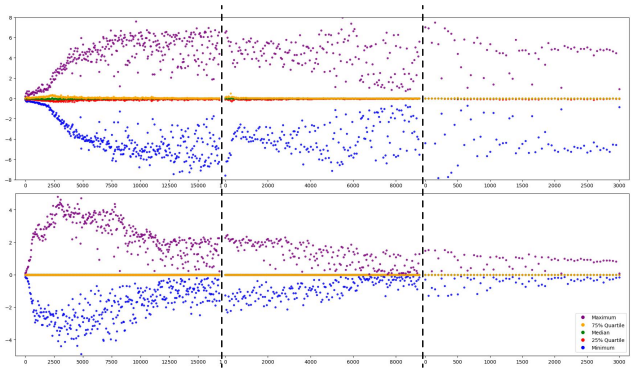

Figure 2. Evolution of the distribution of gradient values of CLAM during CL for: (top) the attention network, (bottom) the classifier network. The data points are aggregated where each dot represents the mean of 50 consecutive training steps. The five colored dots at each aggregated step represent the minimum, maximum, and three quartiles (first, second, and third) of the gradient values. The vertical black dotted lines mark the transitions between tasks.

ining the 1st columns of the tables, we find that the performance does not decline much if the original attention layer $\theta_1$ is used to replace the attention layers learned after subsequent tasks (i.e., used with classifier $\phi_t$). Furthermore, if the classifier after each task is replaced with the original $\phi_1$, there is still a significant decline in Task 1 accuracy (in the 1st row of the tables). These two observations suggest that *the drift of the fine-tuned attention network is the main cause of the performance decline of attention MIL in CL*. Fig. 1 (2-4) illustrate the drift of CLAM's attention during subsequent CL sessions, where the attention on the tumor region (outlined region) is reduced after each CL session.

### 3.3. Analysis of Parameter Update in MIL

To explore the underlying mathematical mechanisms of catastrophic forgetting, we examine the gradient shifts in the attention network and classifier in the MIL model during training updates. As a foundational example, we employed the ABMIL [9] model, where the forward propagation is:

$$f_\phi = \boldsymbol{\phi}^\top \mathbf{H}\mathbf{a}, \tag{3}$$

where $\boldsymbol{\phi} = [\phi_1, \cdots, \phi_d]^\top$ represents the logit layer of the binary classifier, $\mathbf{a} = [a_1, \cdots, a_n]^\top$ are the attention scores from Eq. (2), and $\mathbf{H} = [\mathbf{h_1}, \cdots, \mathbf{h_n}]$ are the feature vectors corresponding to patches in the WSI. In the binary classification setting, the model's loss function is

$$L = \log(1 + e^{-yf_\phi}). \tag{4}$$

From this, the gradients w.r.t. each parameter in the model can be derived (see supplemental). The squared norm of the gradient w.r.t. each classifier weight $\phi_j$ is:

$$\left(\frac{\partial L}{\partial \phi_j}\right)^2 = y^2\sigma^2(-yf_\phi)z_j^2 \tag{5}$$

where $z_j = \sum_{i=1}^n a_i H_{ij}$ represents the $j$-th element of the aggregated feature vector $\mathbf{z}$ and $H_{ij}$ is the $ij$-th element of $\mathbf{H}$. Similarly, the squared norm of the gradient w.r.t. each attention score $a_i$ is:

$$\left(\frac{\partial L}{\partial a_i}\right)^2 = y^2\sigma^2(-yf_\phi)(\boldsymbol{\phi}^\top \mathbf{h}_i)^2. \tag{6}$$

Through Eq. (5) and (6), we analyze the magnitude of the gradients with respect to different parameters in the attention network and the classifier. The main difference between the two equations are the terms $z_j^2$ and $\boldsymbol{\phi}^\top \mathbf{h}_i$. For the gradient of the classifier in Eq. (5), since the normalized attention $\mathbf{a}$ sums to 1 and $\mathbf{H}$ is fixed for an image, then $z_j$ is upper-bounded, $\max z_j = \max_i H_{ij}$. Thus each update step of $\phi_j$ is always constrained within a specific range, which leads to the the classifier's parameters only gradually updating during training. In the context of CL, when there is a significant shift in the data stream, like tasks switch, the classifier only undergoes slow updates, thereby retaining more of the previous knowledge.

In contrast, the attention gradient in (Eq. (6)) directly reflects the importance the patch $\mathbf{h}_i$ to the classifier $\boldsymbol{\phi}$, and this gradient is unbounded – as the classifier becomes more confident ($\|\boldsymbol{\phi}\|^2$ increases), then the attention gradients increase. Therefore, when there is a significant change in the data stream during CL, i.e., $\mathbf{h}_i$ undergoes drastic changes, the gradient of the attention layer will also fluctuate significantly, especially on newly-seen patches with high logit scores ($\boldsymbol{\phi}^\top \mathbf{h}_i$), leading to catastrophic forgetting due to rapid model updates. Furthermore, if the newly seen patch with a high logit score is not informative or noisy, then the resulting large attention gradient makes the attention network overfit to this patch, causing generalization problems.

In summary, the gradients of the classifier are bounded,

while those in the attention layer (and the preceding attention network) are not. Thus, the attention network will be updated faster than the classifier, i.e., the loss decreases more when updating the attention network. Fig. 2 shows the distribution of gradient values of the attention network and the classifier of CLAM as it evolves over successive training steps in CL. During training on the second and third tasks, the maximum and minimum values of the gradient of the classifier gradually shrink and stabilize, but the gradient of the attention network keeps oscillating in a higher range. In the traditional image classification, Kessler et al. [10] found that the performance of CL largely depends on different classification heads across tasks. However, for MIL, we find that switching classification heads will no longer be effective, and our decoupling experiments in Sec. 3.2 further validate this theoretical finding.

# 4. Methodology

In this section, using our analysis in the previous section, we propose two methods for CL of attention-based MIL.

## 4.1. Attention Knowledge Distillation

Inspired by Sec. 3.3, we propose a plug-and-play method called *attention knowledge distillation* (AKD). Compared to the commonly used logits distillation in current CL approaches [2, 12], we focus on distillation of the attention layers since they drift faster in the MIL setting. Moreover, AKD can be combined with logits distillation to further enhance model performance. Previous CL works used the output logit of the classifier $f_\phi(P)$ from previous tasks to mimic the original response for old samples. Therefore, their objective function for distillation is:

$$\mathcal{L}_{logits} = \mathbb{E}_{\mathbf{z}\sim\mathcal{D}_{t-1}}\left[D_{KL}(f_{\phi_{t-1}}(\mathbf{z}) \parallel f_{\phi_t}(\mathbf{z}))\right], \quad (7)$$

where $\phi_{t-1}$ are the optimal parameters at the end of task $t-1$, $\phi_t$ are the current parameters being optimized, and $D_{KL}$ is the KL divergence.

AKD directly optimizes the attention network using attention scores instead of the output logits of the model, reducing the interference caused to the classification layer. Specifically, consider bag feature $\mathbf{z} = \sum_n a_n \mathbf{h}_n$, AKD aims to ensure that after the model learns a new task, it retains the relative order of patch attention scores when inferring on previous tasks. In other words, after completing training on a new task, patches from previous tasks with previously high attention scores should continue to have high attention scores, and those with low scores should maintain relatively low scores. Therefore, we use KL divergence to enforce this relationship of the attention scores:

$$\mathcal{L}_{attn} = \mathbb{E}_{\mathbf{H}\sim\mathcal{M}}\left[D_{KL}(f_{\theta_{t-1}}(\mathbf{H}) \parallel f_{\theta_t}(\mathbf{H}))\right], \quad (8)$$

where $\theta_{t-1}$ is the optimal attention network at the CL session $t-1$, and $\mathbf{H}$ are patch features sampled from the memory pool $\mathcal{M}$. Note that $\mathcal{A} = f_{\theta_t}(\mathbf{H})$ represents the output produced by the attention network, which may encompass more than just the raw attention scores. Depending on the specific architecture of the network, the output can include various feature vectors derived from the attention process. For instance, $\mathcal{A} = \{QK, QKV\}$ in the case of self-attention in TransMIL.

## 4.2. Pseudo-Bag Memory Pool

Traditional replay-based methods typically store thousands of samples in the memory pool – each sample occupies very little space, usually sized at 224x224 pixels, and having a large number of samples ensures that during replay training the model does not overfit. However, in WSI tasks, the entire dataset contains only a few hundred images due to the high costs of data collection and patient privacy concerns. Moreover, each WSI encompasses tens of billions of pixels, making it impractical to store too many raw WSIs directly in the memory pool.

Recent research indicates that attention MIL models primarily utilize a small subset of patches from a WSI when calculating the bag feature $\mathbf{z}$ [27, 30]. This suggests that the original slide can be distilled by removing low-contribution patches: $\mathbf{P} = \{\mathbf{p}_n\}_{n=1}^N \rightarrow \hat{\mathbf{P}} = \{\mathbf{p}_n\}_{n=1}^K$, where $K \ll N$. Yan et al. [27] demonstrated that the top few dozen patches with the highest attention scores account for over 90% of the model's attention. Therefore, one possible approach is to use patches with high attention scores (top-$K$) to approximate the original bag feature:

$$\hat{\mathbf{z}} = \sum\nolimits_{n=1}^{K} a_n \mathbf{h}_n \approx \mathbf{z} = \sum\nolimits_{n=1}^{N} a_n \mathbf{h}_n. \quad (9)$$

Thus, instead of storing the entire slide, we store the distilled slide (pseudo-bag) in the memory pool $\mathcal{M} = \{(\hat{\mathbf{P}}_i, y_i)\}$ to approximate the data distribution of the WSI memory pool $\mathcal{M} = \{(\mathbf{P}_i, y_i)\}$. While using only the top-K patches is a tempting approach, it has been shown that other patches can also contribute valuable information [29].

Therefore, we propose the MaxMinRand strategy to distill the essential information from the WSI:

- **MaxMinRand**: selects $K/2$ patches randomly and selects $K/4$ patches with the highest and the lowest attention scores, respectively.

We also consider other methods and will present them in Sec. 5.4. We adopt the reservoir sampling method [23] to update the slide-level memory pool. The reservoir sampling method can guarantee each slide-level sample shares the same probability of being stored and removed in the memory pool for the current task.

## 4.3. Framework Overview

The model's overall architecture, as depicted in Figure 3, follows a structured, task-oriented data flow. Input data is

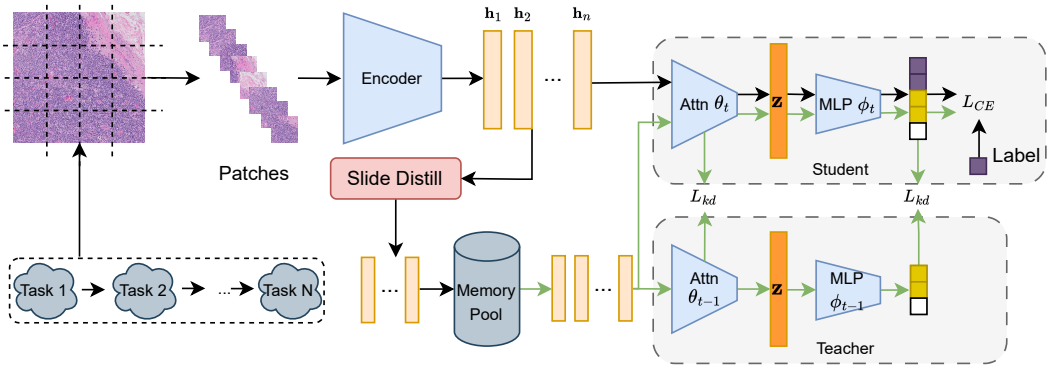


Figure 3. Overview of our framework architecture. The black arrows denote the forward pass for current data, including patch-wise slide processing, feature encoding, and memory pool updates. Orange rectangles represent patch-level features, while deep orange rectangles indicate bag-level feature representations. The green paths indicate the knowledge distillation process between teacher and student networks, incorporating both attention-based and logit-based distillation losses alongside cross-entropy supervision.

sequentially fed into the model in alignment with the progression of tasks. At the conclusion of each task, a subset of distilled samples is retained in a buffer through the PMP (Pseudo-Bag Memory Pool) method. When training transitions to the next task, the model is provided with data from both the current task and the buffered samples. For data from the current task, the model engages in regular parameter updates, as our experiments suggest that interference from prior tasks can impede effective adaptation to new data. During the replay of buffered samples from previous tasks, the model applies AKD to mitigate forgetting of prior knowledge. The final loss function, integrating these components, is formulated as follows:

$$\mathcal{L} = \mathcal{L}_{CE} + \alpha\mathcal{L}_{attn} + \beta\mathcal{L}_{logits}, \tag{10}$$

where $(\alpha, \beta)$ are hyperparameters that apply only to the data in the buffer. The complete process is outlined in Alg. 1.

**Algorithm 1** Overview

**Input:** Attention network $f_\theta$, classifier $f_\phi$, number of tasks $T$, training set $\mathcal{D} = \{\{(\mathbf{H}_i^t, y_i^t)\}_{i=1}^{n_t}\}_{t=1}^{T}, y_i^t \in \mathcal{Y}$, memory pool $\mathcal{M}$.
**Initialize:** $f_\theta, f_\phi, \mathcal{M} = \emptyset$
**for** $t = 1$ to $T$ **do**
  Obtain current task data $\mathcal{D}_t = \{(\mathbf{H}_i^t, y_i^t)\}_{i=1}^{n_t}$.
  Combine current task and memory pool $\bar{\mathcal{D}} = \mathcal{D}_t \cup \mathcal{M}$.
  **for** sample in $\bar{\mathcal{D}}$ **do**
    Obtain $(\mathbf{H}, y)$ from sample
    Calculate attention outputs: $\mathcal{A}, \boldsymbol{z} = f_\theta(\mathbf{H})$.
    $\mathcal{L}_{CE} = \mathcal{L}_{CE}(f_\phi(\boldsymbol{z}), Y)$.
    **if** $(\mathbf{H}, y) \in \mathcal{M}$ **then**
      Obtain previous output $(\tilde{\mathcal{A}}, \tilde{f}_\phi(\boldsymbol{z}))$ from sample.
      $\mathcal{L}_{attn} = \mathcal{L}_{KL}(\mathcal{A}, \tilde{\mathcal{A}})$ (Eq. 8).
      $\mathcal{L}_{logits} = \mathcal{L}_{KL}(f_\phi(\boldsymbol{z}), \tilde{f}_\phi(\boldsymbol{z}))$
    **else**
      $\mathcal{L}_{attn} = 0, \mathcal{L}_{logits} = 0$.
    **end if**
    $\mathcal{L} = \mathcal{L}_{CE} + \alpha\mathcal{L}_{attn} + \beta\mathcal{L}_{logits}$.
    $(\theta, \phi) = (\theta, \phi) + \eta\nabla_{\theta,\phi}(\mathcal{L})$
    $(\bar{\mathbf{H}}, y, \bar{\mathcal{A}}, \bar{f}_\phi(\boldsymbol{z})) = Distill(\mathbf{H}, y, \mathcal{A}, f_\phi(\boldsymbol{z}))$
    $\mathcal{M} \leftarrow reservoir(\mathcal{M}, (\bar{\mathbf{H}}, y, \bar{\mathcal{A}}, \bar{f}_\phi(\boldsymbol{z})))$.
  **end for**
**end for**

## 5. Experiments

In this section we present experiments demonstrating the efficacy of our MIL methods on CL for WSIs.

### 5.1. Setup

**Datasets.** For class incremental learning (CIL), we utilized a skin cancer dataset [5], as well as three publicly available datasets for WSI analysis: Camelyon16 [4, 13], TCGA-LUNG [17], and TCGA-RCC [25]. These datasets represent diverse challenges in histopathology and support the evaluation of our method's ability to generalize across different cancer types and data distributions. Details about the datasets and preprocessing steps are in the Supplemental.

We utilized two primary groups of datasets for class-incremental learning tasks. First, we followed [5], which divides the skin cancer dataset into three class-incremental tasks. Each task within this group comprises two distinct class labels. The second group (denoted as Camelyon-TCGA) consists of the Camelyon16, TCGA-LUNG, and TCGA-RCC datasets, organized sequentially into three tasks based on dataset source: Camelyon16 $\rightarrow$ TCGA-LUNG $\rightarrow$ TCGA-RCC. We employed cross-validation for Camelyon-TCGA. This structured progression enables the model to incrementally learn from different domains, demonstrating its ability to generalize across varied types of cancer and histopathological data.

**Models and CL methods.** We employed CLAM [15] and TransMIL [20] as baseline MIL models. CLAM builds upon ABMIL by incorporating a multiclass categorization framework, making it particularly suitable for our experimental setup involving multiple cancer types. In contrast, TransMIL uses self-attention mechanisms instead of the

gated-attention modules that our theoretical framework assumes. Despite this architectural difference, we apply our AKD to TransMIL to study the generality of our AKD.

We compare performance with state-of-the-art CL methods for image classification, including experience replay (ER) [18], DER++ [2] and LwF [12], and a MIL-specific method, MICIL [5]. We also conduct joint training (JT) of the model using a single dataset of merged tasks, which is considered as the upper-bound performance of CL, and fine-tuning (FT) on each task dataset sequentially, which is the baseline performance.

In the Skin Cancer dataset, each slide contains only a few hundred to a few thousand patches. Thus during CL on this dataset, we do not use our PMP to distill slides, and thus only apply our AKD or other CL methods. For replay-based methods, we set the memory pool size to 42 and use the parameters recommended in their papers for other methods.

On the Camelyon-TCGA, to ensure training stability, we use a balanced memory pool, which maintains an equal number of samples for each class in the memory pool, applying reservoir sampling within each class. Each WSI comprises over 10k patches, presenting a significant memory challenge for CL methods. To manage this, we evaluate replay-based methods with different memory settings of 5, 10, and 30 WSIs for comprehensive comparison.

**Evaluation.** To evaluate the performance of models across CL tasks, we primarily used balanced average accuracy (AACC) to measure the overall performance across all tasks. We also report backward transfer (BWT) and the intransigence measure (IM) to quantify the ability to retain and adapt. See the Supplemental for more details.

### 5.2. Comparison with CL methods

First, we conduct experiments on the Skin Cancer dataset, and the results are presented in Tab. 2. MICIL w/ ER is MICIL with experience replay. All CL methods significantly outperform the fine-tuning approaches. Among these methods, our approach achieves AACC performance that most closely reaches the theoretical upper bound of joint training. Furthermore, the results for BWT and IM indicate that our method keeps the effectiveness in mitigating catastrophic forgetting while facilitating continual adaptation to new information.

The results on Camelyon-TCGA are presented in Tab. 3. Traditional CL methods face considerable challenges on this dataset, which includes over 2,000 large-scale WSIs. These methods struggle to perform effectively, particularly when applied to the TransMIL model, with some CL approaches even underperforming compared to the baseline fine-tuning method. This discrepancy arises because, despite their improved memory retention, traditional CL methods often fall short in adapting to new tasks, as reflected in their BWT and IM scores. In contrast, our method successfully addresses these limitations by integrating our AKD and PMP. This approach mitigates forgetting while also enhancing the model's ability to learn new tasks, resulting in performance gains of up to 37% over other methods when applied to the CLAM model.

Finally, we note that despite the architectural differences in attention between CLAM and TransMIL, our experimental results indicate that TransMIL also works effectively with our proposed AKD, achieving strong outcomes across all evaluated metrics. This demonstrates the robustness and adaptability of the AKD method, even when applied to models with varying attention mechanisms.

### 5.3. Visualizations

Fig. 1 visualizes the heatmap of CLAM attention during different CL sessions. The model's attention shifts when an MIL model is trained sequentially from different tasks. We visualized a sample in the Task 1 test set. The model should ideally focus on the tumor region. However, during fine-tuning, the model tends to reduce its attention to the tumor region and pay much more attention to the normal region. In contrast, for our method, the tumor region is nearly white in the difference heatmap, maintaining the original attention on the important tumor areas. Moreover, the attention changes brought by our method are much lower than those of the fine-tuning method.

### 5.4. Ablation Study

As introduced in Sec. 4.2, we explored several strategies for constructing the pseudo-bag representation. Here, we compare the performance of MaxMinRand with the following alternatives:

- **Random**: randomly selects $K$ patches.
- **Max**: selects $K$ patches with the highest attention scores.
- **MaxMin**: selects $K/2$ patches with the highest attnetion scores, and $K/2$ with the lowest attention scores.
- **MaxRand**: selects $K/2$ patches each from the **Random** strategy and **Max** strategy.

Tab. 4 summarizes the results of these different strategies. Surprisingly, both Random-based and MaxMin-based strategies achieve comparable performance, suggesting that AKD does not heavily rely on patches with the highest attention scores. Instead, AKD appears to emphasize the global attention distribution.

We also investigated the contribution of each component in our method to the overall results. As shown in Tab. 5, our proposed PMP method achieves a 37% and 5% improvement over the basic ER approach using entire WSI, without requiring additional storage. While the AKD method contributes a 40% and 8% improvement. The study clearly underscores the effectiveness of our method, as it enhances performance efficiently without imposing further storage

Table 2. CL performance of CLAM and TransMIL on Skin Cancer dataset. The best performances are highlighted as bold.

| | *CLAM* | | | *TransMIL* | | |
|---|---|---|---|---|---|---|
| **Methods** | **AACC ↑** | **BWT↑** | **IM↓** | **AACC ↑** | **BWT↑** | **IM↓** |
| Joint training | 0.7311 | - | - | 0.7647 | - | - |
| Fine-tuning | 0.3167 | -0.8701 | -0.1942 | 0.3000 | -0.9239 | -0.1771 |
| LwF | 0.3333 | -0.8823 | -0.2339 | 0.3000 | -0.9239 | -0.1923 |
| MICIL | 0.3000 | -0.9239 | -0.1923 | 0.3000 | -0.8944 | -0.2087 |
| ER | 0.5388 | -0.3869 | -0.0942 | 0.6050 | -0.4041 | -0.1356 |
| DER++ | 0.5343 | -0.4931 | -0.1604 | 0.5329 | -0.5124 | -0.1356 |
| MICIL w/ ER | 0.5285 | -0.4523 | -0.1275 | 0.6275 | -0.3705 | -0.1356 |
| Ours | **0.5926** | -0.4056 | -0.1604 | **0.6418** | -0.3741 | -0.1523 |

Table 3. CL performance of CLAM and TransMIL on Camelyon-TCGA dataset. The best performances are highlighted as bold. The standard deviations of BWT and IM metrics can be found in the supplementary material.

| | | | *CLAM* | | | *TransMIL* | | |
|---|---|---|---|---|---|---|---|---|
| **CL Type** | **Method** | **Memory Size** | **AACC ↑** | **BWT ↑** | **IM ↓** | **AACC ↑** | **BWT ↑** | **IM ↓** |
| **Baselines** | Joint training | - | 0.858±0.016 | - | - | 0.818±0.027 | - | - |
| | Fine-tuning | | 0.296±0.014 | -0.865 | -0.014 | 0.290±0.002 | -0.751 | 0.024 |
| **Regularization** | LwF | - | 0.295±0.013 | -0.865 | -0.013 | 0.296±0.010 | -0.747 | 0.021 |
| | MICIL | | 0.295±0.008 | -0.866 | -0.013 | 0.294±0.011 | -0.760 | 0.014 |
| **Rehearsal** | ER | 5 WSIs | 0.294±0.010 | -0.864 | -0.012 | 0.296±0.008 | -0.748 | 0.020 |
| | DER++ | | 0.301±0.010 | -0.857 | -0.013 | 0.288±0.021 | -0.752 | 0.025 |
| | MICIL w/ ER | | 0.289±0.013 | -0.869 | -0.010 | 0.290±0.009 | -0.759 | 0.018 |
| | Ours | | **0.657±0.032** | -0.329 | -0.017 | **0.374±0.060** | -0.635 | 0.018 |
| | ER | 10 WSIs | 0.352±0.063 | -0.785 | -0.017 | 0.297±0.012 | -0.746 | 0.021 |
| | DER++ | | 0.372±0.070 | -0.760 | -0.020 | 0.299±0.015 | -0.738 | 0.024 |
| | MICIL w/ ER | | 0.298±0.006 | -0.863 | -0.015 | 0.302±0.016 | -0.746 | 0.016 |
| | Ours | | **0.729±0.041** | -0.217 | -0.059 | **0.394±0.085** | -0.595 | 0.024 |
| | ER | 30 WSIs | 0.494±0.058 | -0.565 | -0.011 | 0.308±0.011 | -0.728 | 0.021 |
| | DER++ | | 0.449±0.048 | -0.645 | -0.020 | 0.315±0.028 | -0.739 | 0.006 |
| | MICIL w/ ER | | 0.308±0.021 | -0.835 | -0.006 | 0.298±0.019 | -0.739 | 0.024 |
| | Ours | | **0.754±0.029** | -0.177 | -0.058 | **0.489±0.059** | -0.460 | 0.018 |

Table 4. Ablation study of different distillation methods for AKD on CLAM and TransMIL in Camelyon-TCGA dataset.

| **Method** | **CLAM AACC** | **TransMIL AACC** |
|---|---|---|
| Random | 0.695±0.010 | 0.391±0.065 |
| MaxMin | 0.679±0.025 | 0.391±0.054 |
| Max | 0.595±0.121 | 0.340±0.008 |
| MaxRand | 0.705±0.021 | 0.395±0.036 |
| MaxMinRand | 0.729±0.041 | 0.394±0.085 |

Table 5. Ablation study on our components using CLAM on Camelyon-TCGA dataset.

| **Method** | **CLAM AACC** | **TransMIL AACC** |
|---|---|---|
| ER | 0.296±0.014 | 0.290±0.002 |
| AKD | 0.692±0.037 | 0.373±0.045 |
| PMP | 0.672±0.025 | 0.348±0.052 |
| PMP+AKD | 0.729±0.041 | 0.394±0.085 |

demands, demonstrating a significant advancement in optimizing resource-constrained learning environments.

# 6. Conclusion

In this paper, our analysis reveals that the main cause of catastrophic forgetting in attention MIL models is the degradation of bag-level feature representations due to changes in the attention module. To address this, we propose a pseudo-bag memory pool (PMP) method to enable replay-based CL for MIL and an attention knowledge distillation (AKD) technique to preserve the attention relationships and maintain representation quality. Experiments on WSI datasets demonstrate the effectiveness of PMP and AKD in significantly improving CL performance for traditional MIL architectures. This work establishes an important foundation for further research on CL for gigapixel images and weakly-supervised learning in real-world clinical applications.

**Limitations.** The current limitation is that our methods still need to store past patch features. While storing features largely reduces privacy risks compared to raw images, it may still pose privacy concerns. In the future, we will explore more privacy-friendly data replay methods.

## References


[1] Rahaf Aljundi, F. Babiloni, Mohamed Elhoseiny, Marcus Rohrbach, and T. Tuytelaars. Memory aware synapses: Learning what (not) to forget. *European Conference on Computer Vision*, 2017. 2

[2] Pietro Buzzega, Matteo Boschini, Angelo Porrello, Davide Abati, and Simone Calderara. Dark experience for general continual learning: a strong, simple baseline. *Advances in neural information processing systems*, 33:15920–15930, 2020. 3, 5, 7

[3] Péter Bándi, Oscar Geessink, Quirine Manson, Marcory Van Dijk, Maschenka Balkenhol, Meyke Hermsen, Babak Ehteshami Bejnordi, Byungjae Lee, Kyunghyun Paeng, Aoxiao Zhong, Quanzheng Li, Farhad Ghazvinian Zanjani, Svitlana Zinger, Keisuke Fukuta, Daisuke Komura, Vlado Ovtcharov, Shenghua Cheng, Shaoqun Zeng, Jeppe Thagaard, Anders B. Dahl, Huangjing Lin, Hao Chen, Ludwig Jacobsson, Martin Hedlund, Melih Çetin, Eren Halıcı, Hunter Jackson, Richard Chen, Fabian Both, Jörg Franke, Heidi Küsters-Vandevelde, Willem Vreuls, Peter Bult, Bram van Ginneken, Jeroen van der Laak, and Geert Litjens. From detection of individual metastases to classification of lymph node status at the patient level: The camelyon17 challenge. *IEEE Transactions on Medical Imaging*, 38(2):550–560, 2019. 1

[4] Babak Ehteshami Bejnordi, Mitko Veta, Paul Johannes van Diest, Bram van Ginneken, Nico Karssemeijer, Geert Litjens, Jeroen A. W. M. van der Laak, , and the CAMELYON16 Consortium. Diagnostic Assessment of Deep Learning Algorithms for Detection of Lymph Node Metastases in Women With Breast Cancer. *JAMA*, 318(22):2199–2210, 2017. 1, 6

[5] Pablo Meseguer Esbri, Rocío del Amor, and Valery Naranjo. Micil: Multiple-instance class-incremental learning for skin cancer whole slide images. *Artificial intelligence in medicine*, 152:102870, 2024. 1, 2, 3, 6, 7

[6] Zeyu Gao, Anyu Mao, Yuxing Dong, Jialun Wu, Jiashuai Liu, Chunbao Wang, Kai He, Tieliang Gong, Chen Li, and Mireia Crispin-Ortuzar. Accurate spatial quantification in computational pathology with multiple instance learning. *medRxiv*, pages 2024–04, 2024. 2

[7] Raia Hadsell, Dushyant Rao, Andrei A Rusu, and Razvan Pascanu. Embracing change: Continual learning in deep neural networks. *Trends in cognitive sciences*, 24(12):1028–1040, 2020. 1

[8] Yanyan Huang, Weiqin Zhao, Shujun Wang, Yu Fu, Yuming Jiang, and Lequan Yu. Conslide: Asynchronous hierarchical interaction transformer with breakup-reorganize rehearsal for continual whole slide image analysis. *IEEE International Conference on Computer Vision*, 2023. 1, 2, 3

[9] Maximilian Ilse, Jakub M. Tomczak, and M. Welling. Attention-based deep multiple instance learning. *International Conference on Machine Learning*, 2018. 2, 3, 4

[10] Samuel Kessler, Adam D. Cobb, Tim G. J. Rudner, Stefan Zohren, and Stephen J. Roberts. On sequential bayesian inference for continual learning. *Entropy*, 25(6):884, 2023. 5

[11] 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. *Proceedings of the national academy of sciences*, 114(13):3521–3526, 2017. 1, 2

[12] Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2016. 3, 5, 7

[13] Geert Litjens, Peter Bandi, Babak Ehteshami Bejnordi, Oscar Geessink, Maschenka Balkenhol, Peter Bult, Altuna Halilovic, Meyke Hermsen, Rob Van de Loo, Rob Vogels, et al. 1399 h&e-stained sentinel lymph node sections of breast cancer patients: the camelyon dataset. *GigaScience*, 7 (6):giy065, 2018. 1, 6

[14] David Lopez-Paz and Marc'Aurelio Ranzato. Gradient episodic memory for continual learning. *Advances in neural information processing systems*, 30, 2017. 3

[15] Ming Y Lu, Drew FK Williamson, Tiffany Y Chen, Richard J Chen, Matteo Barbieri, and Faisal Mahmood. Data-efficient and weakly supervised computational pathology on whole-slide images. *Nature Biomedical Engineering*, 5(6):555–570, 2021. 1, 2, 3, 6

[16] Arun Mallya and Svetlana Lazebnik. Packnet: Adding multiple tasks to a single network by iterative pruning. *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2017. 3

[17] Cancer Genome Atlas Research Network et al. Comprehensive molecular profiling of lung adenocarcinoma. *Nature*, 511(7511):543, 2014. 6

[18] M. Riemer, Ignacio Cases, R. Ajemian, Miao Liu, I. Rish, Y. Tu, and G. Tesauro. Learning to learn without forgetting by maximizing transfer and minimizing interference. *International Conference on Learning Representations*, 2018. 7

[19] Andrei A. Rusu, Neil C. Rabinowitz, Guillaume Desjardins, Hubert Soyer, James Kirkpatrick, Koray Kavukcuoglu, Razvan Pascanu, and Raia Hadsell. Progressive neural networks. *arXiv preprint arXiv: 1606.04671*, 2016. 3

[20] Zhuchen Shao, Hao Bian, Yang Chen, Yifeng Wang, Jian Zhang, Xiangyang Ji, et al. Transmil: Transformer based correlated multiple instance learning for whole slide image classification. *Advances in Neural Information Processing Systems*, 34:2136–2147, 2021. 2, 3, 6

[21] Shusuke Takahama, Y. Kurose, Yusuke Mukuta, Hiroyuki Abe, A. Yoshizawa, T. Ushiku, M. Fukayama, M. Kitagawa, Masaru Kitsuregawa, and Tatsuya Harada. Domain adaptive multiple instance learning for instance-level prediction of pathological images. *IEEE International Symposium on Biomedical Imaging*, 2023. 1

[22] Gido M Van de Ven and Andreas S Tolias. Three scenarios for continual learning. *arXiv preprint arXiv:1904.07734*, 2019. 1

[23] Jeffrey Scott Vitter. Random sampling with a reservoir. *ACM Trans. Math. Softw.*, 11:37–57, 1985. 5

[24] Liyuan Wang, Xingxing Zhang, Hang Su, and Jun Zhu. A comprehensive survey of continual learning: theory, method and application. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024. 1

[25] Cancer Genome Atlas Research Network Analysis working group: Baylor College of Medicine Creighton Chad J. 1 2 Morgan Margaret 1 Gunaratne Preethi H. 1 3 Wheeler David A. 1 Gibbs Richard A. 1, BC Cancer Agency Gordon Robertson A. 4 Chu Andy 4, Broad Institute Beroukhim Rameen 5 6 Cibulskis Kristian 6, Brigham & Women's Hospital Signoretti Sabina 7 54 59, Brown University Vandin Hsin-Ta Wu Fabio 8 8 Raphael Benjamin J. 8, University of Texas MD Anderson Cancer Center Verhaak Roel GW 9 Tamboli Pheroze 10 Torres-Garcia Wandaliz 9 Akbani Rehan 9 Weinstein John N. 9, Memorial Sloan-Kettering Cancer Center Reuter Victor 11 Hsieh James J. 12 Rose Brannon A. 11 Ari Hakimi A. 12 Jacobsen Anders 13 Ciriello Giovanni 13 Reva Boris 13, National Cancer Institute Ricketts Christopher J. 14 Marston Linehan W. agibbs@ bcm. edu 14 c, University of California Santa Cruz Stuart Joshua M. 15, Chapel Hill Kimryn Rathmell W. 16 University of North Carolina, et al. Comprehensive molecular characterization of clear cell renal cell carcinoma. *Nature*, 499(7456):43–49, 2013. 6

[26] Yunyang Xiong, Zhanpeng Zeng, Rudrasis Chakraborty, Mingxing Tan, G. Fung, Yin Li, and Vikas Singh. Nyströmformer: A nyström-based algorithm for approximating self-attention. *AAAI Conference on Artificial Intelligence*, 2021. 3

[27] Renao Yan, Qiehe Sun, Cheng Jin, Yiqing Liu, Yonghong He, Tian Guan, and Hao Chen. Shapley values-enabled progressive pseudo bag augmentation for whole slide image classification. *IEEE Transactions on Medical Imaging*, 2023. 5

[28] Cui Yufei, Ziquan Liu, Xiangyu Liu, Xue Liu, Cong Wang, Tei-Wei Kuo, Chun Jason Xue, and Antoni B Chan. Bayesmil: A new probabilistic perspective on attention-based multiple instance learning for whole slide images. In *The Eleventh International Conference on Learning Representations*, 2023. 1

[29] Hongrun Zhang, Y. Meng, Yitian Zhao, Yihong Qiao, Xiaoyun Yang, S. Coupland, and Yalin Zheng. Dtfd-mil: Double-tier feature distillation multiple instance learning for histopathology whole slide image classification. *Computer Vision and Pattern Recognition*, 2022. 1, 2, 5

[30] Yunlong Zhang, Honglin Li, Yuxuan Sun, Sunyi Zheng, Chenglu Zhu, and Lin Yang. Attention-challenging multiple instance learning for whole slide image classification, 2023. 2, 5