

Self-Supervision Enhances Instance-based Multiple Instance Learning Methods in Digital Pathology: A Benchmark Study

Ali Mammadov^{1,2*}, Loïc Le Folgoc¹, Julien Adam², Anne Buronfosse²,
Gilles Hayem², Guillaume Hocquet², Pietro Gori¹

¹Telecom Paris, Institut Polytechnique de Paris, France.

²Groupe Hospitalier Paris Saint-Joseph, France.

*Corresponding author. E-mail: ali.mammadov@ip-paris.fr;

Abstract

Multiple Instance Learning (MIL) has emerged as the best solution for Whole Slide Image (WSI) classification. It consists of dividing each slide into patches, which are treated as a bag of instances labeled with a global label. MIL includes two main approaches: instance-based and embedding-based. In the former, each patch is classified independently, and then the patch scores are aggregated to predict the bag label. In the latter, bag classification is performed after aggregating patch embeddings. Even if instance-based methods are naturally more interpretable, embedding-based MILs have usually been preferred in the past due to their robustness to poor feature extractors. However, recently, the quality of feature embeddings has drastically increased using self-supervised learning (SSL). Nevertheless, many authors continue to endorse the superiority of embedding-based MIL. To investigate this further, we conduct 710 experiments across 4 datasets, comparing 10 MIL strategies, 6 self-supervised methods with 4 backbones, 4 foundation models, and various pathology-adapted techniques. Furthermore, we introduce 4 instance-based MIL methods never used before in the pathology domain. Through these extensive experiments, we show that with a good SSL feature extractor, simple instance-based MILs, with very few parameters, obtain similar or better performance than complex, state-of-the-art (SOTA) embedding-based MIL methods, setting new SOTA results on the BRACS and Camelyon16 datasets. Since simple instance-based MIL methods are naturally more interpretable and explainable to clinicians, our results suggest that more effort should be put into well-adapted SSL methods for WSI rather than into complex embedding-based MIL methods.

Keywords: Whole Slide Image Classification, Self-Supervised Learning, Multiple Instance Learning, Digital Pathology

1 Introduction

Whole Slide histopathology Image (WSI) analysis has become an increasingly common tool for disease diagnosis in digital pathology [1]. However, the gigapixel size of WSIs, makes the manual analysis very time-consuming and presents significant challenges for conventional Deep Learning (DL) methods [2, 3], as they are not designed to support such large images. To address that, a simple approach involves dividing the WSI into smaller patches that DL methods can easily handle. Then, features or predictions from a patch-level encoder/classifier are aggregated to get the slide-level prediction [4, 5]. Nonetheless, this method requires very expensive patch-level annotations, which are not always available. Please note that a naive assignment of the slide label to all patches might be clinically incorrect, since the tissue section characterizing a disease might only occupy a small fraction of the slide, while all other patches should be considered as healthy.

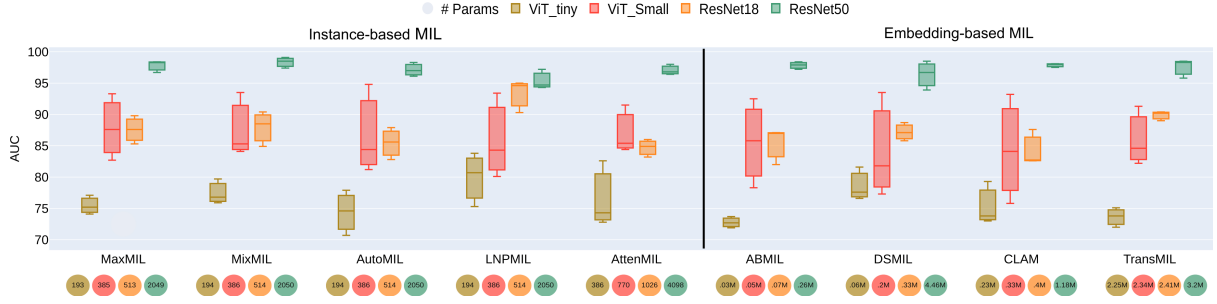


Fig. 1 Self-supervision (SSL) closes the gap between instance- and embedding-based multiple-instance learning (MIL) methods in whole slide image (WSI) classification. Instance-based MIL methods, when combined with robust SSL feature extractors, are on par or even outperform complex, state-of-the-art embedding-based methods in WSI classification. The Y-axis represents the AUC scores of different MIL methods for four backbones (ViT.tiny, ViT.Small, ResNet18 and ResNet50) on the Camelyon16 dataset (at a resolution 20x for ResNet50 and 10x for the other backbones). For each backbone and MIL method, we show the box-plot of 3 self-supervised pre-trainings: DINO [6], MOCO-V3 [7] and Barlow Twins [8].

1.1 Multiple-Instance Learning (MIL)

To tackle this challenge, Multiple-Instance Learning (MIL) methods, coupled with Deep Learning Feature Extraction (FE), have emerged for WSI classification using only slide (i.e., weak) labels. This approach is considered the most prominent solution in the field of histopathology, where FE avoids the costly and experience-based feature engineering part, while MIL eliminates the need for patch-level (or pixel-level) annotations [9, 10]. Under MIL formulation, each WSI is treated as a "bag" containing multiple instances in the form of patches, which are embedded with Convolutional Neural Network (CNN) or Vision Transformer (ViT) [11] backbones. The bag is labeled positive (i.e., diseased) if at least one of its patches is positive, or negative if all patches are negatives [12, 13]. In general, the existing methodologies follow a two-step pipeline: 1) feature extraction from individual patches, and 2) MIL aggregation through a pooling operation to predict the slide label [14–19]. Many works in the literature focus on the second part of the pipeline, developing various aggregation methods that can be categorized into two groups: *instance-based* and *embedding-based* methods. Instance-based methods use an instance-level classifier, which predicts a score for each patch. Then, these scores are aggregated via a MIL-based pooling operator to make the final prediction for the entire slide. Common pooling operators include average-pooling (*MeanMIL*) and max-pooling (*MaxMIL*) [20, 21]. Instance-based MIL methods are actively used in several domains [22, 23], and in particular in sound event detection [20, 24]. However, these methods were only used in few and early works in the field of Digital Pathology, such as in [12]. Even if these methods are naturally interpretable and easily explainable, they highly depend on the quality of the embedding. To increase reliability, researchers proposed to aggregate features instead of scores, moving the classification head after the pooling. These are called embedding-based methods.

The existing literature has proposed several approaches for feature aggregation based on: deep self-attention mechanism [13, 15], graph convolutional networks [14, 25, 26], clustering [19, 27], transformer [17] and additional training [18, 28]. These pooling mechanisms are usually more complex (more parameters) than instance-based ones. On the one hand, this means that the model has a greater capability of learning how to correctly aggregate the features and thus might have a greater prediction power. On the other hand, interpretability and explainability can decrease¹ and at the same time computational complexity and overfitting can increase (i.e., the number of needed training samples increases). Until recently, most of the researchers used ImageNet pre-trained models to extract features. However, as shown in [29, 30], these models might not be optimal for histopathology images and may produce poor representations (i.e., features) due to the domain gap between natural and medical images. This might explain why most of the early, ImageNet-based works reported that embedding-based MIL models outperformed instance-based ones [17, 31–33]. In recent years, to enhance feature quality, the advent of large, weakly annotated WSI datasets has led many researchers to investigate self-supervised learning methods.

1.2 Self Supervised Learning (SSL)

SSL is a paradigm whose goal is to learn useful and relevant representations for downstream tasks by leveraging pretext tasks from unlabeled data. It has shown promise in improving the performance of image

¹even though attention and self-attention mechanisms can be exploited in that regard

classification [6–8, 34–36]. In digital pathology, SSL has been actively used in recent works. Notably, in [16] authors use SimCLR [36] to train a feature extractor. Other authors use MoCo [7] in [37, 38], DINO [6] in [39], BYOL [34] in [40] and MoCo V3 [41] in [42]. Some recent papers have even proposed novel self-supervised learning methods specifically tailored for digital pathology and achieved new state-of-the-art results. In [43], authors apply the DINO method at different scales for extracting hierarchical features, providing a more comprehensive representation of the histopathology images. Another significant contribution is the Semantically Relevant Contrastive Learning (SRCL) method introduced by [44]. Based on MoCoV3, SRCL additionally compares relevance between instances to mine more positive pairs. An original self-supervised method based on an auto-encoder was explored by [45], which learns to reconstruct the masked borders of an image from a given center. In [46], authors enhance the BYOL method by incorporating a clustering constraint. Additionally, [47] and [48] propose different and innovative contrastive learning-based approaches. Another line of research focuses on the use of SSL at the slide level, as proposed by [49, 50]. All of these approaches have shown that self-supervision outperforms ImageNet pre-training on both patch-level and slide-level downstream tasks.

1.3 Existing Limitations

Even if recent works provided very insightful results for WSI classification using SSL methods, their experiments and comparison present some limitations. They either use a single SSL method [40, 51], or compare few SSL methods [39], or multiple SSL methods but with *different* backbones [52], or they use a *single* backbone for all methods [17]. Furthermore, most of these works focus on a few embedding-based MIL methods, using at most one/two instance-based methods [17, 37, 46, 51]. Another limitation is that some works use few datasets (and usually with a low clinical complexity², like Camelyon16) for binary classification only [46].

To address these limitations, we present here a fair, complete and thorough comparison on *four* different datasets, for both binary and multi-class classification, that present an increasing level of clinical complexity. We investigate the importance of all the key elements of the WSI classification pipeline, testing: 4 different *feature extractors (backbones)*, 6 *SSL* strategies (plus a pre-training of Image-Net), 4 *foundation models* and 10 *MIL* methods. This research fills a significant gap in the literature of WSI classification by providing a comprehensive study and showcasing the potential of simple and interpretable instance-based MIL methods, when combined with robust SSL methods.

1.4 Our Contributions and Main Results

- We conduct a large-scale study on 4 datasets, with 7 different pre-training methods, 4 *foundation models*, 4 different backbone models, and 10 multiple-instance learning methods for a total of 710 different configurations.
- We propose simple instance-based MIL approaches based on pooling mechanisms borrowed from the sound event detection field which, to the best of our knowledge, have never been used in pathology.
- We demonstrate that simple instance-based MIL methods, when combined with robust SSL feature extractors, are on par or even outperform complex embedding-based methods in WSI classification, *whatever* the employed backbone model (CNN or ViT), see Fig.1.
- Using our newly proposed instance-based MIL methods, we achieve new state-of-the-art (SOTA) results in BRACS [42] and Camelyon16 [28], and we are on par with current SOTA methods in TCGA-NSCLC [53].

We also share our code, pre-trained models, extracted features, hyperparameters and valuable insights for WSI classification in https://github.com/mammadov7/wsi_classification_pipeline.

2 Methods

2.1 MIL problem formulation

Let $X_i = \{x_{ij}\}_{j=1\dots K_i}$ be the whole slide image of subject i (i.e., bag) and $x_{ij} \in \mathbb{R}^d$ its j -th patch (i.e. instance). We have $N \in \mathbb{Z}^+$ bags in our training set and $K_i \in \mathbb{Z}^+$ instances for each bag i (here we assume that K_i can vary across bags). There is a single classification label y_i for each bag (binary or multi-class) and we assume that each instance has also its own label y_{ij} (same classes as for the bag), but it is unknown during training.

Following [13, 54], there are two main approaches: instance-level and embedding-level. We define here a

²clinical complexity is measured in terms of average AUC score.

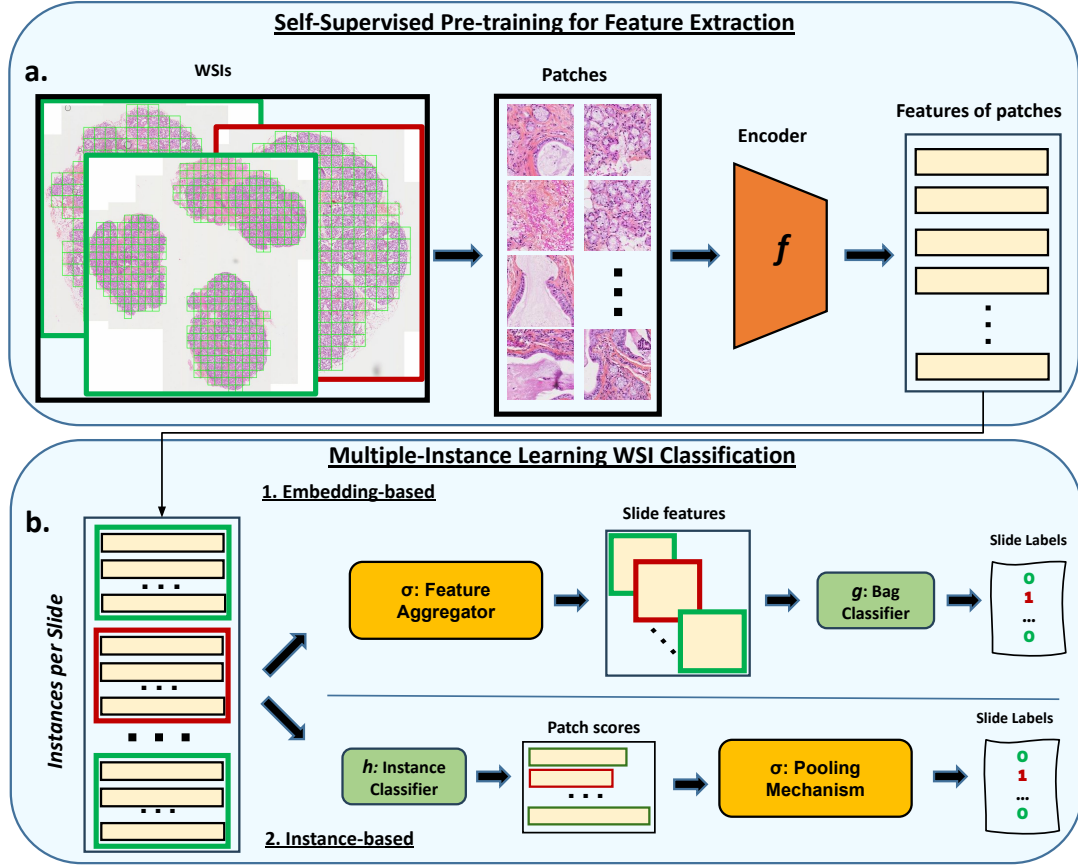


Fig. 2 Pipeline of the WSI classification. At the top, a) represents a self-supervised pre-training of an encoder f on patches from all slides. Once pre-trained, the encoder f extracts a vector of features (i.e., representation) per patch. Representations of patches of the same slide are then stacked together. At the bottom, b.1) Embedding-based MIL: features from the same slide are aggregated together using a feature aggregator σ into a bag-level representation. A bag/slide classifier g is then trained on bag/slide representations to predict the slide label. b.2) Instance-based MIL: an instance classifier h assigns scores (i.e., class probability) to each patch, which are then pooled using a pooling mechanisms σ to predict the slide label.

generic framework (see Fig.2) well adapted to both. Let $f(\cdot) : \mathbb{R}^d \rightarrow \mathbb{R}^p$ be a parametric function, here a deep learning encoder, that maps the instances to a feature space and σ a MIL pooling operator (i.e., a symmetric, permutation-invariant function). The final bag-score $S(X)$ is:

$$S(X_i) = g(\sigma(h(f(x)))) \quad (1)$$

where $g(\cdot)$ and $h(\cdot)$ are bag-level and instance-level classifiers respectively. In the instance-level approach, we only use $h(\cdot)$, which computes a score (class probability) for each instance, fixing $g(\cdot)$ to the identity function. Instead, in the embedding-level approach, we use $g(\cdot)$, which computes a score at the bag level, setting $h(\cdot)$ to the identity mapping.

2.2 Instance-based MIL pooling operators

In this class of methods, the pooling mechanism is used to aggregate patch scores. The two most used MIL pooling operators σ are the $\max(\cdot)$, *MaxMIL*, and the $\text{mean}(\cdot)$, *MeanMIL*, which have served as baseline in many WSI classification works, such as [13, 16–18, 38, 54, 55]. However, Max-pooling may not capture sufficient information from a single patch and Average-pooling may gather excessive information from all patches, leading to suboptimal results. Unfortunately, one does not necessarily know *a priori* which pooling mechanism would be the most suitable for a given dataset. In the sound event detection domain, authors have proposed simple generalizations [21], notably: Learned-Norm Pooling (LNP) [56], Mix-pooling [57], Auto-pooling [24] and Attention-pooling [20]. These approaches involve learning a set of weights or norms associated with each patch’s contribution. The pooling mechanism is not fixed but is adaptively adjusted during the training process based on the significance of each patch for the classification task. This adaptability allows the model to assign more weight to informative patches and less to less relevant ones.

2.3 Proposed Instance-based MIL pooling operators

Let $y_k = h(f(x_k)) \in [0, 1]$ the predicted probability (score) of patch k , we define:

MixMIL (Mixed-Pooling) [57]: $\sigma = \alpha \max_{k=1 \dots K} y_k + (1 - \alpha) \sum_{k=1}^K y_k$ where $\alpha \in [0, 1]$ is a trainable parameter.

AutoMIL (Auto-pooling) [24]: $\sigma = \sum_{k=1}^K y_k \left(\frac{\exp(\alpha \cdot y_k)}{\sum_{j=1}^K \exp(\alpha \cdot y_j)} \right)$ where $\alpha \in [0, \infty)$ is a trainable parameter.

LNPMIL (Softmax-pooling) [56]: $\sigma = \left(\frac{1}{K} \sum_{k=1}^K |y_k|^p \right)^{\frac{1}{p}}$; $p = 1 + \log(1 + e^{\tilde{p}})$ where $\tilde{p} \in \mathbb{R}$ is a trainable parameter.

AttenMIL (Attention-pooling) [20]: $\sigma = \sum_{k=1}^K y_k \hat{w}_k$ with $\hat{w} = \text{softmax}(w_k)$ where the weights w_k , one for each patch, are learned with a dedicated layer in the network.

2.3.1 Multi-class Instance-based MIL

Here, we propose a simple extension to multiple classes of the MaxMIL pooling operator.

Multi-class MaxMIL: for each instance x_{ik} , the instance-level classifier outputs a score vector $h(x_{ik})$ whose components are the (scalar) scores $s_{ik}^{(c)} \in [0, 1]$ for each class c . Assuming 3 classes, we have:

$$h(x_{ik}) = \left(s_{ik}^{(1)}, s_{ik}^{(2)}, s_{ik}^{(3)} \right)$$

where we make the hypothesis that $c = 1$ refers to the normal/benign class while $c = 2, 3$ refer to two different tumoral/pathological classes (as in BRACS [58]). Cancerous slides may exhibit large portions of normal/benign tissues, which is usually easier to classify. As a result, they will be probably assigned a high score $s_{ik}^{(1)}$ and thus, simply applying max pooling over the patch scores of all classes, it would probably select a normal tissue patch even in a cancerous slide. To avoid that, we propose to perform the aggregation by first selecting the instance with the highest score among the second and third classes:

$$m = \arg \max_k (s_{ik}^{(2)}, s_{ik}^{(3)})$$

Then, the bag-level prediction is made by taking the maximum score among all the scores of the instance m :

$$y_i = \arg \max \left(s_{im}^{(1)}, s_{im}^{(2)}, s_{im}^{(3)} \right)$$

In BRACS, the three classes are: Benign Tumor, Atypical Tumor, and Malignant Tumor.

Multi-class MixMIL (Mixed-Pooling), we use the aforementioned strategy only for the MaxMIL part of MixMIL.

2.4 Embedding-based MIL pooling operators

Differently from instance-based MIL methods, here the pooling mechanisms aggregate features, and not scores (i.e., scalars). Embedding-based MIL methods are usually more complex (i.e., more parameters) since they need to effectively capture and aggregate information from high-dimensional vectors [13–19, 28]. In our study, we employ the following four and frequently used state-of-the-art methods:

ABMIL [13]: is an attention-based MIL. It employs trainable attention layers to attribute an attention score to each patch representation, which are then aggregated via a weighted sum for the final slide classification.

$$\sigma = \sum_{k=1}^K a_k f_k \text{ where } f_k = f(x_k) \text{ and } a_k = \frac{\exp(w^T \tanh(V f_k^T))}{\sum_{j=1}^K \exp(w^T \tanh(V f_j^T))} \quad (2)$$

The parameters w and V are learned with trainable attention layers.

CLAM [15]: also uses an attention-based aggregation mechanism, and proposes instance-level clustering over the patches to improve the feature space.

DSMIL [16]: is an hybrid, dual-stream method that jointly learns an instance and a bag classifier. First, it detects the critical patch $m = \arg \max_x (h(f(x_k)))$ as in MaxMIL. Then, it transforms each instance $f_k = f(x_k)$ into two vectors: query $q_k = W_q f_k$ and information $v_k = W_v f_k$, where W_q and W_v are two weight matrices. Query vectors are used to compute distances between every patch and the critical one:

$U(f_k, f_m) = \text{softmax}(q_k \cdot q_m)$, where \cdot denotes the inner product. Similarly to ABMIL, the bag aggregator σ is a weighted sum. However, now the weights are based on the similarity of the features with respect to the critical sample: $\sigma = \sum_k U(f_k, f_m) v_k$. It is used to compute the final score $S(X) = \frac{1}{2}(h(f(x_m)) + g(\sigma))$, which is an average between the instance and bag classifiers and where h and g are linear classifiers.

TransMIL [17]: is a transformer-based MIL that considers the correlation between different patches (tokens) using Transformers to investigate morphological and spatial details and it uses self-attention mechanisms for aggregation. In particular, σ is a module, called TPT module, with two Transformer layers and a position encoding layer, where Transformer layers are designed for aggregating morphological information and a Pyramid Position Encoding Generator (PPEG) is designed for encoding spatial information.

DAMIL: [59] is Deep Attention Multiple Instance Survival Learning. It leverages a siamese Multiple Instance Fully Convolutional Network (MI-FCN) to learn features from phenotype clusters derived from WSIs. The framework employs attention-based MIL pooling to aggregate these features into patient-level representations, enhancing interpretability and performance.

DTFDMIL: [18] is Double-Tier Feature Distillation Multiple Instance Learning. This method introduces the concept of "pseudo-bags" which are smaller subsets of patches created from the original slides. The framework employs a two-tier MIL model: the first tier processes these pseudo-bags to distill key features, while the second tier aggregates these features to make predictions at the slide level.

2.5 Self Supervised Pre-training of f

The encoder $f(\cdot)$ is usually pre-trained using self-supervision (SSL). Different strategies exist, including contrastive learning, clustering, knowledge distillation, and information maximization methods. Among the existing SSL methods, we opt for the following six, highly-used and state-of-the-art approaches:

SimCLR [36] learns representations by maximizing agreement between differently augmented views of the same data sample and by minimizing agreement between different samples, via a contrastive loss in the latent space. It comprises four key components: a data augmentation module, a base encoder (*e.g.*, ResNet), a projection head (*e.g.*, multi-layer perceptron (MLP)), and a contrastive loss function (InfoNCE loss). Within this framework, every image in the batch undergoes a sequential process. First, it is transformed twice (*i.e.*, augmented) to create two distinct augmented views. These augmented views are then encoded using the base encoder and projected into a latent space. Subsequently, the contrastive InfoNCE loss is applied to encourage the attraction of positive pairs (representations originating from the same image) and the repulsion of negative pairs (representations of different images) within the latent space. Upon the completion of training, we keep the encoder for downstream tasks.

BYOL [34] uses both online and target neural networks. Its architecture is asymmetrical, with the online network consisting of three stages: encoder, projector, and predictor. The target network comprises just two stages, the encoder, and the projector, but with different weights. In this approach, only the online network is trained and the target network parameters are updated by an exponential moving average of the online parameters. During training, based on an augmented view of an image, the online network is trained to predict the target network representation of the same image under a different augmented view. For a given image, BYOL will produce two different augmented views, $v1$ and $v2$. First, $v1$ will pass through the online network and $v2$ through the target network creating $r1$, $r2$. Then, vice-versa, to symmetrize the pipeline, $v2$ passes through the online network and $v1$ through the target, resulting in $r3$, $r4$. Finally, it minimizes a loss $L = \text{sim}(r1, r2) + \text{sim}(r3, r4)$, where sim is a l_2 -normalized mean squared error function. At test time, only the encoder of the online network is used for feature extraction.

Barlow Twins [8] share similarities with *SimCLR* in that it learns representations of images under different augmented views passing through the same network. The originality of this approach is the non-contrastive loss, which requires no negative pairs or large batches. It measures the cross-correlation matrix between the embeddings obtained from a network fed with distorted versions of the same input image, and tries to make this matrix close to the identity. It makes the embedding invariant to the distortions applied, so that the outputs of the network are as similar as possible for the distorted views of the same input while ensuring that the individual components of these embeddings are not redundant.

DINO [6], similarly to *BYOL*, employs the teacher-student distillation paradigm with two parallel networks and an update mechanism with exponential moving averages. The main differences are that it uses a ViT backbone and Cross Entropy metric as a training loss. Moreover, DINO generates a set of *local* views, comprising several augmented small crops (each less than 50% of the image area), and *global* views (two large crops, each covering more than 50% of the image). All crops pass through the student network, while only the global views pass through the teacher network. Lastly, to avoid collapse

in the presence of a momentum teacher, it applies the centering and sharpening to the momentum teacher outputs. Centering prevents one dimension to dominate but encourages collapse to the uniform distribution, while sharpening has the opposite effect.

MoCo v3 [41] combines *BYOL*’s teacher-student distillation architecture with *SimCLR*’s contrastive learning and *DINO*’s Vision Transformer backbone, enhancing representation learning by leveraging the best of all three approaches.

MAE [35] is a masked autoencoder that randomly masks 75% of the input and then learns to reconstruct the original image from the unmasked parts of the image. It uses an architecture with an encoder operating on unmasked patches and a lightweight decoder reconstructing the full image using latent representations and mask tokens. This efficient design allows for training large encoders with reduced computing and memory, where only a lightweight decoder handles the complete set of patches. MAE uses as optimization loss the mean squared error (MSE) between the reconstructed and original images.

2.6 Pathology Adapted SSL Methods

In the literature, there are very few SSL methods conceived and developed specifically for histopathology images. Furthermore, they either propose well-adapted augmentations or slight modifications of existing methods (which have been mainly developed for natural images). In this section, we describe three recent works that characterize well this class of methods.

PathAug [52], in this work authors proposed three novel image augmentation techniques, specifically developed for histopathological images. The first technique involves vertical flipping of the input images. The second technique uses different fields-of-view by taking patches from various magnifications of the WSIs. The third technique is based on stain augmentation and normalization (RandStainNA), as proposed by [60].

SRCL [40] (semantically-relevant contrastive learning) draws inspiration from two well-known contrastive learning methods: SimCLR [36] and MoCo v3 [41]. Authors propose to select additional semantically relevant positive pairs using a cosine similarity metric, thereby increasing the diversity of positive samples. This approach uses three parallel paths: online, target, and shared target branches, all employing the CTransPath architecture. A memory bank, updated iteratively, stores features from the shared target branch. The online branch is learned via gradient-based optimization, while the target branch is updated through an exponential moving average of the online network parameters. Each input sample generates three views (two augmented, one original), which are passed through the three branches, with the online branch features serving as anchor. By calculating cosine similarities between the anchor and the features in the memory bank, they identify top matches as new positive samples, enhancing the contrastive loss function by maximizing the similarity between these samples and the anchor.

CluBYOL [46] Cluster Bootstrap Your Own Latent (CluBYOL) is a self-supervised learning approach. It extends the original BYOL [34] framework to better capture the underlying structure of clustered data. The main idea is to integrate clustering into the self-supervised learning process, as in SwAV [61], allowing the model to learn more meaningful and discriminative features by leveraging the natural grouping within the data, which is adapted for different tissue types in pathology.

2.7 Foundation Models

Foundation models are large models that are pre-trained on large-scale datasets and can be adapted for a wide range of downstream tasks. For our study, we included one general vision model and three pathological foundation models, as zero-shot feature encoders $f(\cdot)$.

DINOv2 proposed by [62] has emerged as an improved version of the original DINO model. DINOv2 combines the loss functions of DINO [6] and iBOT [63] with the centering of SwAV [61]. We use the ViT_Small model, which is pre-trained on 142 million images.

PathAugFM, proposed by [52], involves pre-training the ViT_Small model over 200 epochs on a dataset containing 19 million pathological patches. The authors utilize the DINO framework coupled with pathology-adapted augmentations, which are detailed in the previous section.

CTransPath from [40], uses a hybrid backbone. This architecture is obtained by replacing the patch embedding component of Swin Transformers with a CNN layer. *CTransPath* is trained on a dataset consisting of 15 million pathological patches with the SRCL framework.

UNI, as introduced by [64], represents a general-purpose self-supervised model tailored for pathology applications. UNI’s training involves pre-training the ViT_Large architecture with the DINOv2 framework, using a vast dataset comprising more than 100 million tissue patches. These patches are extracted

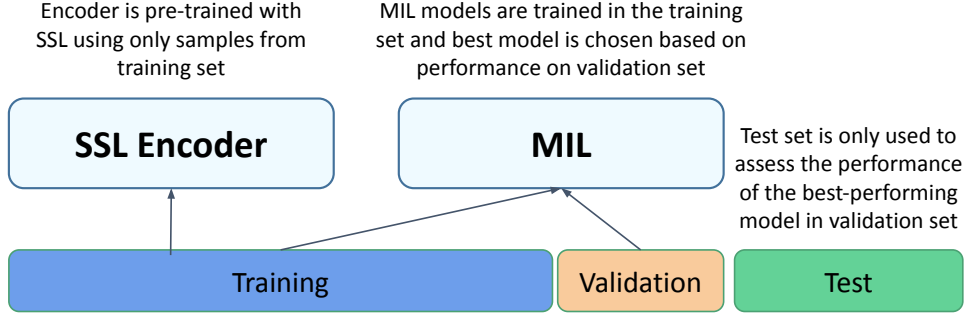


Fig. 3 Visual explanation of the proposed validation pipeline.

from a collection of over 100,000 diagnostic hematoxylin and eosin-stained WSIs, covering a diverse array of 20 major tissue types.

3 Experiments

3.1 Datasets and Experimental Setup

We evaluate the proposed MIL and SSL methods on 4 different histopathological datasets – Camelyon16 [65], The Cancer Genome Atlas non-small cell lung cancer³ (TCGA-NSCLC), VisioMel⁴ and BReAst Carcinoma Subtyping (BRACS) [58] – that: a) have an increasing clinical complexity, b) comprise both binary and multiple-class classification tasks and c) present different levels of class imbalance.

Camelyon16 is a 2-class dataset for the detection of metastases in breast cancer. It includes 400 slides: 239 are normal tissue slides and 160 tumor slides. The official dataset is already split into training and test sets. For CAMx10, following [16], we randomly split the official training set into training and validation sets at a 9:1 ratio. We tune all hyper-parameter values on the validation set. Instead, for CAMx20, we implement a 3-fold cross-validation on the training set.

TCGA-NSCLC is a 2-class, balanced lung cancer subtyping dataset with a total of 997 WSIs: 482 for Lung Squamous Cell Carcinoma (LUSC) and 515 for Lung Adenocarcinoma (LUAD). The WSIs are randomly divided into training, validation, and test sets with a ratio of 65:15:20.

VisioMel is used for the detection of the primary melanoma relapse. It is a 2-class, highly-imbalanced dataset that comprises 1342 cases: 213 positive cases and 1139 negative cases. As the test set is not yet released, we randomly split the official training set into training and validation sets at a 9:1 ratio. We report performances on the validation set.

BRACS is a 3-class imbalanced dataset. It contains 547 WSI for breast carcinoma subtyping out of which 265 are Benign Tumor, 89 are Atypical Tumor, and 193 are Malignant Tumor cases. We use the official split of 395, 65 and 87 samples for train, validation, and test sets, respectively.

Our main evaluation metric is the AUC score which is resilient to class imbalance effects. We select the best-performing models on the validation set and report their AUC scores on the test sets, see Fig. 3. Only for VisioMel, where the test set has not been released, we report the AUC scores on the validation set but, to limit over-optimistic results (and overfitting), we select the model from the last epoch, rather than the one with the best validation loss.

3.2 Pre-processing

We follow the CLAM’s pre-processing pipeline [15] for all datasets. Following prior literature, we cut all WSIs into 256x256 non-overlapping patches solely from foreground tissue regions at x10 magnification, the most commonly used resolution. Only for Camelyon16, we extract patches both at x10 (*CAMx10*) and x20 (*CAMx20*) magnification for a more comprehensive analysis. This process resulted in a total of 4.6 million patches for CAMx20, 0.6 million patches for CAMx10, 2.7 million for TCGA-NSCLC, 1.4 million for BRACS and 2.7 million patches for VisioMel. We chose 256x256 over 224x224, as it results in fewer patches per slide, which helps reduce computational overhead.

³<http://www.cancer.gov/tcga>

⁴<https://www.drivendata.org/competitions/148/visiomel-melanoma>

Table 1 Self-Supervised Pre-Training Hyperparameters, where P is probability and S is scale

| Training | | Augmentations | |
|---------------------------------|---------------------------|------------------------|--|
| <i>Optimizer</i> | LARS | <i>Crop</i> | $S = (0.5, 1.0)$ |
| <i>Scale Loss</i> (λ) | 0.1 | <i>ColorJitter</i> | $S = (0.4, 0.4, 0.2, 0.1)$, $P = 0.8$ |
| <i>Scheduler</i> | Cosine Annealing | <i>Grayscale</i> | $P = 0.2$ |
| <i>Base Learning Rate</i> | 0.3 | <i>GaussianBlur</i> | $S = (0.1, 2.0)$, $P = 1.0$ |
| <i>Weight Decay</i> | $1e-6$ | <i>Solarization</i> | $P = 0.0$ |
| <i>Batch Size</i> | 256×4 , (4 GPUs) | <i>Horizontal Flip</i> | $P = 0.5$ |

3.3 Self-Supervised Learning Pre-Training Details

We include in our study ImageNet pre-training as a baseline and 6 different SSL methods: SimCLR [36], MoCoV3 [41], MAE [35], DINO [6], BYOL [34] and Barlow Twins [8]. With the exception of MAE [35] (compatible only with ViT), these methods can be used with arbitrary backbones. We experiment with four of the most used backbones in histopathological studies: *ResNet18* (11.7M parameters), *ResNet50* (25.5M parameters) [66], *ViT-Tiny* (5.8M parameters) and *ViT-Small* (22.2M parameters) [11]. All SSL models are initialized using the Imagenet-1K weights. Please note that the foundation models CTransPath and UNI are based on two different architectures. In particular, UNI uses a ViT-Large backbone, which has 307 millions parameters, and it is thus much larger than the backbones used in this work. All pre-trainings are done using the *solo-learn* library from [67] for 200 epochs. Regarding the SSL hyperparameters, we keep the same values as in the original papers, except for the crop scale factor, which is reduced to range $\{0.5, 1\}$, to preserve the size of cells, as larger cells might be tumor cells. Additionally, for the learning rate (lr) of each SSL method, as well as for the temperature parameter τ of SimCLR, we perform a grid-search, choosing as optimal value the one giving the best average AUC score across all tested MIL models in the validation set. Following [67], we use LARS (respectively AdamW) for CNN (resp. ViT) backbones, and we use a warm-up phase of 10 epochs with a cosine annealing learning rate scheduler. Furthermore, we select the biggest *batchsize* that fits in the GPU memory of our computational cluster as previous works consistently show that larger batch sizes yield better results. We choose the maximum batch size that fits in our GPU memory for each experiment. *ResNet18* and *ViT-Tiny* backbones are trained on 4 NVIDIA V100 GPUs with *batchsize* 256 per GPU, resulting in a global *batchsize* of $4 \times 256 = 1024$. In order to keep the same *batchsize* also for larger *ResNet50* and *ViT-Small* backbones, we used 8 NVIDIA V100 GPUs with a *batchsize* of 128 for each, $8 \times 128 = 1024$. The hyperparameters for all the combinations of SSL methods, backbones, and datasets can be found in our Github repository⁵ in the following directory: *pre-train/scripts/pretrain/custom*. In Table 1, as an example, we show the hyperparameters used for the Barlow Twins SSL method with ResNet18 backbone on the TCGA-NSCLC dataset. The config file path is *(pre-train/scripts/pretrain/custom/tcga/barlow-res.yaml)* in our repository.

3.4 Pathology Adapted SSL methods details

As proposed by [52], we apply pathology-specific augmentations, vertical flipping and stain augmentations, to three SSL methods (Barlow Twins, MOCOv3, BYOL) to assess whether these adaptations can further enhance the results. However, we exclude multiple magnification levels due to their high computational cost. All the other training components and hyperparameters remain unchanged, as described in the previous subsection. Unfortunately, we could not include the *SRCL* method, due to the lack of code in the source repository. For CluBYOL, we use the same hyper-parameters as in the original paper [46] and recompute the statistics on the validation set.

3.5 Foundation Models Details

We perform no training or fine-tuning of the foundation models. Instead, we download the pre-trained weights from their original repositories and load them into the corresponding backbones, as detailed in Section 2.6. We then use these models directly to extract features from the patches, which we further use for MIL trainings.

3.6 MIL Training Details

We adopt DSMIL’s code [16] as a base for our training and evaluation pipeline. Then, for each MIL method, we replace the model, optimizers and other training components based on their official codes.

⁵https://github.com/mammadov7/wsi_classification_pipeline

Table 2 Camelyon16 dataset at 10x. First and second-best AUC scores in bold and underlined respectively. The best average AUC scores (per MIL) in red. ImageNet-based results are not averaged. A Welch T-Test was applied to compare the best performing method (Ref.) with the others, where (*) indicates statistically significant differences ($p < 0.05$) and "N/S" denotes not significance. The *DAMIL* method introduces additional complexity because the K-Means algorithm is applied independently on each slide.

| Arch. | Pretrain. | MeanMIL | MaxMIL | MixMIL | AutoMIL | LNPMIL | AttenMIL | ABMIL | DSMIL | CLAM | TransMIL | DTFDMIL | DAMIL | Avg. |
|-----------|----------------|----------|-----------------|-----------------|-------------|-----------------|-------------|-----------------|-----------------|----------|-----------------|-------------|----------|----------|
| # Params. | (193) | (193) | (194) | (194) | (194) | (386) | (386) | (0.03M) | (0.06M) | (0.23M) | (2.25M) | (0.18M) | (0.02M) | (-) |
| ViT-Tiny | ImageNet | 58.1 | 69.2 | 72.5 | 71.6 | 71.5 | 73.9 | 65.5 | 70.5 | 71.2 | 66.7 | 71.4 | 58.6 | 68.4±5.0 |
| | Barlow T. | 58.5 | 74.1 | 79.7 | 77.9 | 83.8 | <u>82.6</u> | 71.9 | 77.6 | 79.3 | 75.1 | 70.7 | 69.6 | 75.1±6.6 |
| | MOCov3 | 51.8 | 75.2 | 75.9 | 70.7 | 80.7 | 74.3 | 72.7 | 76.6 | 73.8 | 72.0 | 80.2 | 71.4 | 72.9±7.1 |
| | SimCLR | 61.0 | 76.0 | 74.2 | 77.4 | 73.8 | 75.4 | 76.9 | 72.8 | 77.5 | 72.7 | 76.4 | 63.2 | 73.1±5.2 |
| | BYOL | 63.0 | 72.1 | 73.7 | 73.8 | <u>82.6</u> | 77.4 | 74.4 | 74.5 | 69.2 | 78.5 | 74.2 | 52.3 | 72.1±7.5 |
| | DINO | 64.5 | 77.1 | 76.8 | 74.6 | 75.3 | 72.8 | 73.7 | 81.6 | 73.0 | 73.8 | 48.2 | 63.7 | 71.3±8.4 |
| | MAE | 50.6 | 57.4 | 59.5 | 54.2 | 61.9 | 66.9 | 72.1 | 67.4 | 64.7 | 65.2 | 71.2 | 56.6 | 62.3±6.5 |
| | Avg. | 58.2±5.3 | 72.0±6.7 | 73.3±6.5 | 71.4±8.1 | 76.4±7.4 | 74.9±4.7 | 73.6±1.7 | 75.1±4.4 | 72.9±4.9 | 72.9±4.0 | 70.1±10.3 | 62.8±6.7 | 71.1±8.1 |
| | <i>p-value</i> | * | N/S | N/S | N/S | Ref. | N/S | N/S | N/S | N/S | N/S | N/S | * | - |
| | # Params. | (385) | (385) | (386) | (386) | (386) | (770) | (0.05M) | (0.2M) | (0.33M) | (2.34M) | (0.23M) | (0.03M) | (-) |
| ViT-Small | ImageNet | 57.4 | 64.8 | 77.2 | 75.4 | 75.0 | 72.6 | 76.1 | 74.9 | 71.8 | 73.6 | 71.1 | 64.4 | 71.2±5.7 |
| | Barlow T. | 50.4 | 87.6 | 85.3 | 81.2 | 84.3 | 85.4 | 78.3 | 77.3 | 84.1 | 84.6 | 85.2 | 64.3 | 79.0±10. |
| | MOCov3 | 58.5 | 93.3 | <u>93.5</u> | 94.8 | 93.4 | 91.5 | 92.5 | <u>93.5</u> | 93.2 | 91.3 | 92.8 | 62.9 | 87.6±12. |
| | SimCLR | 60.5 | 73.2 | 72.5 | 67.5 | 76.9 | 63.5 | 75.5 | 76.1 | 69.2 | 69.0 | 79.0 | 76.3 | 71.6±5.5 |
| | BYOL | 62.9 | 85.9 | 89.0 | 83.3 | 90.8 | 80.1 | 84.4 | 86.4 | 86.8 | 84.8 | 85.7 | 69.3 | 82.4±7.9 |
| | DINO | 63.2 | 82.7 | 84.1 | 84.4 | 80.1 | 84.4 | 85.8 | 81.8 | 75.8 | 82.2 | 78.7 | 63.8 | 78.9±7.4 |
| | MAE | 50.2 | 70.1 | 72.3 | 68.4 | 60.6 | 74.5 | 79.0 | 77.3 | 73.8 | 78.4 | 70.5 | 54.8 | 69.2±8.9 |
| | Avg. | 57.6±5.4 | 82.1±8.1 | 82.8±7.9 | 79.9±9.5 | 81.0±11. | 79.9±9.0 | 82.6±5.7 | 82.1±6.2 | 80.5±8.3 | 81.7±6.9 | 82.0±7.0 | 65.2±6.5 | 78.1±11. |
| | <i>p-value</i> | * | N/S | Ref. | N/S | N/S | N/S | N/S | N/S | N/S | N/S | N/S | * | - |
| | # Params. | (513) | (513) | (514) | (514) | (514) | (1026) | (0.07M) | (0.33M) | (0.4M) | (2.41M) | (0.26M) | (0.03M) | (-) |
| ResNet18 | ImageNet | 56.2 | 66.8 | 73.4 | 70.5 | 67.8 | 72.4 | 72.9 | 60.1 | 62.5 | 70.2 | 73.1 | 64.3 | 67.5±5.4 |
| | Barlow T. | 71.0 | 87.3 | 90.4 | 88.4 | 94.8 | 88.5 | 88.1 | 91.2 | 88.2 | 91.5 | 93.9 | 66.9 | 86.7±8.3 |
| | MOCov3 | 72.3 | 87.6 | 88.5 | 87.9 | 96.0 | 84.9 | 87.0 | 88.7 | 82.6 | 90.2 | 89.2 | 65.0 | 85.0±8.1 |
| | SimCLR | 72.6 | 89.4 | 90.9 | 90.0 | <u>95.1</u> | 91.4 | 86.5 | 90.3 | 93.8 | 92.6 | 81.7 | 62.5 | 86.4±9.3 |
| | BYOL | 60.7 | 73.6 | 74.7 | 77.4 | 74.4 | 75.5 | 74.1 | 73.8 | 75.2 | 75.1 | 78.5 | 53.1 | 72.2±7.1 |
| | DINO | 69.2 | 89.0 | 90.8 | 90.3 | 92.7 | 90.8 | 88.3 | 88.6 | 84.4 | 91.9 | 92.0 | 72.2 | 86.7±7.5 |
| | Avg. | 69.2±4.4 | 85.4±5.9 | 87.1±6.2 | 86.8±4.8 | 90.6±8.2 | 86.2±5.8 | 84.8±5.4 | 86.5±6.4 | 84.8±6.2 | 88.3±6.6 | 87.1±6.0 | 63.9±6.3 | 83.4±9.9 |
| | <i>p-value</i> | * | N/S | N/S | N/S | Ref. | N/S | N/S | N/S | N/S | N/S | N/S | * | - |
| | # Params. | (2049) | (2050) | (2050) | (2050) | (4098) | (4098) | (0.26M) | (4.46M) | (1.18M) | (3.2M) | (0.66M) | (0.14M) | (-) |
| | ImageNet | 64.4 | 83.2 | 77.7 | 77.1 | 72.0 | 72.0 | 77.2 | 80.7 | 86.3 | 78.9 | 77.4 | 63.4 | 76.2±6.8 |
| ResNet50 | Barlow T. | 98.3 | 98.5 | 96.1 | 97.2 | 96.4 | 96.4 | 98.4 | 98.5 | 98.1 | 98.3 | 94.3 | 65.1 | 94.5±9.4 |
| | MOCov3 | 96.7 | 97.4 | 97.0 | 94.3 | 96.8 | 96.8 | 97.2 | 96.7 | 97.5 | 95.8 | 97.1 | 66.7 | 93.9±8.7 |
| | SimCLR | 81.4 | 92.3 | 92.4 | 84.5 | 91.2 | 91.2 | 89.9 | 88.0 | 92.4 | 92.0 | 94.8 | 62.2 | 87.4±8.8 |
| | BYOL | 96.4 | 97.1 | 96.0 | 90.9 | 94.4 | 94.4 | 95.9 | 94.5 | 97.9 | 97.9 | 94.9 | 61.4 | 92.5±10. |
| | DINO | 98.4 | 99.1 | 98.3 | 94.7 | 98.0 | 98.0 | 97.9 | 93.9 | 98.0 | 98.5 | <u>98.8</u> | 64.4 | 94.6±9.7 |
| | Avg. | 94.2±6.5 | 96.9±2.4 | 96.0±2.0 | 92.3±4.4 | 95.4±2.4 | 95.9±3.1 | 94.3±3.6 | 96.8±2.2 | 96.5±2.4 | 96.0±1.7 | 94.0±1.9 | 64.0±1.9 | 92.6±9.7 |
| | <i>p-value</i> | N/S | Ref. | N/S | N/S | N/S | N/S | N/S | N/S | N/S | N/S | N/S | * | - |
| | # Params. | (2049) | (2050) | (2050) | (2050) | (4098) | (4098) | (0.26M) | (4.46M) | (1.18M) | (3.2M) | (0.66M) | (0.14M) | (-) |
| | ImageNet | 64.4 | 83.2 | 77.7 | 77.1 | 72.0 | 72.0 | 77.2 | 80.7 | 86.3 | 78.9 | 77.4 | 63.4 | 76.2±6.8 |
| | Barlow T. | 98.3 | 98.5 | 96.1 | 97.2 | 96.4 | 96.4 | 98.4 | 98.5 | 98.1 | 98.3 | 94.3 | 65.1 | 94.5±9.4 |

Table 3 Camelyon16 dataset at 20x. First and second-best AUC scores are in bold and underlined respectively. The best average AUC scores (per MIL) in red. ImageNet-based results are not averaged. A Welch T-Test was applied to compare the best performing method (Ref.) with the others, where (*) indicates statistically significant differences ($p < 0.05$) and "N/S" denotes not significance. The *DAMIL* method introduces additional complexity because the K-Means algorithm is applied independently on each slide.

| Arch. | Pretrain. | MaxMIL | MixMIL | AutoMIL | LNPMIL | AttenMIL | ABMIL | DSMIL | CLAM | TransMIL | DTFDMIL | DAMIL | Avg. |
|----------|----------------|----------|------------------|----------|----------|----------|----------|----------|------------------|----------|-------------|----------|----------|
| | # Params. | (2049) | (2050) | (2050) | (2050) | (4098) | (0.26M) | (4.46M) | (1.18M) | (3.2M) | (0.66M) | (0.14M) | (-) |
| ResNet50 | ImageNet | 64.4 | 83.2 | 77.7 | 77.1 | 72.0 | 77.2 | 80.7 | 86.3 | 78.9 | 77.4 | 63.4 | 76.2±6.8 |
| | Barlow T. | 98.3 | 98.5 | 96.1 | 97.2 | 96.4 | 98.4 | 98.5 | 98.1 | 98.3 | 94.3 | 65.1 | 94.5±9.4 |
| | MOCov3 | 96.7 | 97.4 | 97.0 | 94.3 | 96.8 | 97.2 | 96.7 | 97.5 | 95.8 | 97.1 | 66.7 | 93.9±8.7 |
| | SimCLR | 81.4 | 92.3 | 92.4 | 84.5 | 91.2 | 89.9 | 88.0 | 92.4 | 92.0 | 94.8 | 62.2 | 87.4±8.8 |
| | BYOL | 96.4 | 97.1 | 96.0 | 90.9 | 94.4 | 95.9 | 94.5 | 97.9 | 97.9 | 94.9 | 61.4 | 92.5±10. |
| | DINO | 98.4 | 99.1 | 98.3 | 94.7 | 98.0 | 97.9 | 93.9 | 98.0 | 98.5 | <u>98.8</u> | 64.4 | 94.6±9.7 |
| | Avg. | 94.2±6.5 | 96.9 ±2.4 | 96.0±2.0 | 92.3±4.4 | 95.4±2.4 | 95.9±3.1 | 94.3±3.6 | 96.8 ±2.2 | 96.5±2.4 | 96.0±1.7 | 64.0±1.9 | 92.6±9.7 |
| | <i>p-value</i> | N/S | Ref. | N/S | N/S | N/S | N/S | N/S | N/S | N/S | N/S | * | - |

We trained each MIL model for 100 *epochs* with a grid search for optimal *learning.rate*. Fig. 4.1 illustrates the applied grid search process. The figure consists of six subplots, each corresponding to a different pre-training SSL method. In each subplot, the Validation AUC scores of 12 MIL methods are plotted on the Y-axis, while the learning rate is represented on the X-axis. For example, in the case of SimCLR, we select 0.0005 as the optimal learning rate for MaxMIL, as it achieves the highest validation score. We use the cosine annealing scheduler, Adam optimizer with a weight decay of 0.00001 and a batch size equal to 1 slide (*i.e.*, bag). For MIL, due to the varying number of patches per slide, the batch size is always 1. This is a standard practice in MIL literature and is followed in all related works. The bag size depends on the number of patches per slide. Since this varies across slides, it is not a fixed hyperparameter. For more details about the other hyper-parameters please refer to the released code.

4 Results and Discussions

In this section, we share our results from all datasets and discuss our findings based on quantitative and qualitative results.

Table 4 Foundation models on Camelyon16 dataset at 10x. First and second-best AUC scores in bold and underlined respectively. Best average AUC score in red.

| Pretrain | Camelyon16 | | | | | | | | | |
|-----------------------|------------|----------|----------|------------------|-------------|----------|----------|----------|------------------|----------|
| | MeanMIL | MaxMIL | MixMIL | AutoMIL | LNPMIL | AttenMIL | ABMIL | DSMIL | CLAM | TransMIL |
| SRCL(CTransPath) | 57.3 | 91.1 | 89.8 | 92.4 | 94.7 | 91.9 | 89.2 | 91.3 | 88.0 | 94.7 |
| PathAugFM(ViT_Small) | 60.2 | 91.5 | 95.3 | 95.8 | 90.9 | 93.6 | 94.4 | 91.7 | 94.7 | 91.4 |
| DINOv2(ViT_Small) | 60.0 | 46.3 | 76.4 | 80.1 | 70.3 | 55.6 | 76.0 | 49.9 | 79.6 | 77.1 |
| UNI(ViT_Large) | 64.5 | 97.7 | 96.8 | 96.8 | <u>98.8</u> | 96.7 | 97.4 | 95.7 | 99.1 | 98.3 |
| Avg. | 60.5±2.6 | 81.6±21. | 89.6±8.0 | 91.3 ±6.7 | 88.7±11. | 84.4±17. | 89.2±8.2 | 82.2±19. | <u>90.4</u> ±7.4 | 90.4±8.0 |

Table 5 AUC scores of different MIL methods with Pathology Adapted SSL methods on Camelyon16 dataset at 10x magnification, *-path* suffix indicates methods with adapted augmentations. The red/green numbers show if adapted augmentations downgraded/improved results: $method_{path} - method$.

| Pretrain | Camelyon16 | | | | | | | | | | |
|--------------------------------|------------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|
| | MeanMIL | MaxMIL | MixMIL | AutoMIL | LNPMIL | AttenMIL | ABMIL | DSMIL | CLAM | TransMIL | Avg. |
| Barlow | 71.0 | 89.3 | 90.4 | 88.4 | 94.8 | 88.5 | 88.1 | 91.2 | 88.2 | 91.5 | 87.9 |
| <i>Barlow</i> _{path} | 69.4-1.6 | 93.5+6.2 | 93.1+2.7 | 94.6+6.2 | 94.9+0.1 | 92.3+3.8 | 87.4-0.7 | 88.9-2.3 | 90.6+2.4 | 88.8-2.7 | 89.4+1.5 |
| MOCov3 | 72.3 | 87.6 | 88.5 | 87.9 | 96.0 | 84.9 | 87.0 | 88.7 | 82.6 | 90.2 | 86.6 |
| <i>MOCov3</i> _{path} | 71.2-1.1 | 92.3+4.7 | 92.4+3.9 | 93.5+5.6 | 91.8-4.2 | 92.6+7.7 | 79.2-7.8 | 85.3-3.4 | 86.5+3.9 | 93.4+3.2 | 87.8+1.2 |
| BYOL | 60.7 | 73.6 | 74.7 | 77.4 | 74.4 | 75.5 | 74.1 | 73.8 | 75.2 | 75.1 | 73.5 |
| <i>BYOL</i> _{path} | 64.9+4.2 | 75.1+1.5 | 78.4+3.7 | 80.2+2.8 | 71.0-3.4 | 82.0+6.5 | 81.1+7.0 | 80.3+6.5 | 80.6+5.4 | 74.8-0.3 | 76.8+3.3 |
| CluBYOL | 58.4 | 89.7 | 89.4 | 90.1 | 91.4 | 70.4 | 74.9 | 89.3 | 85.2 | 89.0 | 82.1 |
| <i>CluBYOL</i> _{path} | 59.8+1.4 | 88.8-0.9 | 88.0-1.4 | 89.0-1.1 | 92.7+1.3 | 86.8+16. | 83.4+8.5 | 80.6-8.7 | 87.3+2.1 | 82.7-6.3 | 83.9+1.8 |

Table 6 Effect on the final AUC score of the *ImageNet* initialization using ResNet18 as backbone. We compare three SSL techniques with all MIL methods on the Camelyon16 dataset at 10x magnification. Each result (Average_{std}) is the average and standard deviation of three different trainings. *IN* suffix stands for *ImageNet* initialization, no suffix indicates random initialization and $\Delta = method_{IN} - method$.

| Arch. | Pretrain. | MeanMIL | MaxMIL | MixMIL | AutoMIL | LNPMIL | AttenMIL | ABMIL | DSMIL | CLAM | TransMIL | Avg. |
|----------|---------------------------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| | Params. | (513) | (513) | (514) | (514) | (514) | (1026) | (0.07M) | (0.33M) | (0.4M) | (2.41M) | (-) |
| ResNet18 | Barlow T. | 71.8±1.5 | 89.3±0.4 | 88.8±1.5 | 90.4±0.9 | 95.1±0.3 | 89.8±0.4 | 86.3±1.4 | 89.6±0.4 | 89.2±2.1 | 89.6±1.4 | 87.7 |
| | Barlow T. <i>IN</i> | 71.0±1.3 | 87.3±1.9 | 90.4±1.3 | 88.4±2.0 | 94.8±0.1 | 88.5±1.9 | 88.1±1.6 | 91.2±1.8 | 88.2±1.4 | 91.5±1.2 | 87.9 |
| | Barlow T. <i>IN</i> - Barlow T. | - 0.8 | - 2.0 | + 1.6 | - 2.0 | - 0.3 | - 1.3 | + 1.8 | + 1.6 | - 1.0 | + 1.9 | + 0.1 |
| | SimCLR | 72.6±0.2 | 89.4±1.3 | 90.6±1.8 | 92.1±2.3 | 95.4±0.5 | 91.9±0.7 | 88.9±2.3 | 91.2±0.3 | 91.5±1.6 | 90.2±2.1 | 89.4 |
| | SimCLR <i>IN</i> | 72.6±0.4 | 89.4±1.4 | 90.9±2.2 | 90.0±0.5 | 95.1±0.5 | 91.4±1.8 | 86.5±1.7 | 90.3±2.2 | 93.8±0.2 | 92.6±1.5 | 89.3 |
| | SimCLR <i>IN</i> - SimCLR | 0.0 | 0.0 | + 0.3 | - 2.1 | - 0.3 | - 0.5 | - 2.4 | - 0.9 | + 2.3 | + 2.4 | - 0.1 |
| | DINO | 69.7±1.0 | 89.1±1.7 | 91.9±0.5 | 91.1±0.8 | 94.6±0.8 | 90.2±0.9 | 87.7±1.3 | 88.6±1.1 | 83.8±0.8 | 92.9±2.1 | 88.0 |
| | DINO <i>IN</i> | 69.2±0.9 | 89.0±2.3 | 90.8±0.6 | 90.3±0.6 | 92.7±1.2 | 90.8±2.0 | 88.3±1.4 | 88.6±1.0 | 84.4±2.0 | 91.9±1.6 | 87.6 |
| | DINO <i>IN</i> - DINO | - 0.5 | - 0.1 | - 1.1 | - 0.8 | - 1.9 | + 0.6 | + 0.6 | 0.0 | + 0.6 | - 1.0 | - 0.4 |

4.1 Camelyon16

Tables 2, 3, 4, 5 present the classification results for Camelyon16 using various backbones, SSL, and MIL methods.

Self-supervised pre-training. Table 2 presents results for ViT-Tiny, ViT-Small, and ResNet18 backbones using data at x10 magnification, while Table 3 presents results for ResNet50 using data at x20 magnification. Across all setups, instance-based MIL methods are on par, or slightly better, than embedding-based MIL methods (on average over all SSL methods), despite having fewer parameters by several orders of magnitude. The best-performing combinations of SSL and MIL is based on instance-based MIL. Notably, the proposed MixMIL method, with a ResNet50 backbone pretrained with DINO on Camelyon16 at x20 magnification, achieves a new SOTA result with a 99.1 AUC score.

Foundation models. Table 4 reports MIL methods’ performance when using foundation models as feature extractors, including the general-purpose DINOv2 model [62] and in-domain CTransPath [44], PathAugFM [52] and UNI [64]. We observe a similar trend between instance-based and embedding-based MIL methods, as with the previous SSL pre-trainings. We can also notice that the results of DINOv2 are significantly lower compared to the other models, which may be related to the domain gap between natural and pathological images.

Pathology-adapted SSL. Table 5 reports the effect of the pathology-adapted SSL methods on the performance of the MIL techniques. We compare three SSL methods (Barlow Twins, MOCov3, BYOL), using the standard augmentations for natural images, as proposed in the original articles, and their histopathological counterparts, denoted with the *-path* suffix, where we use the pathology-specific augmentations proposed in [52]: vertical flipping and stain augmentation. For all methods, we use the ResNet18 backbone. We can see that results can vary quite a lot between standard and histopathological augmentations, going from a difference of -0.9 to a one of +16. However, on average, pathology-adapted augmentations improve performance by 1.2 to 3.3 AUC points across all MIL approaches. Additionally, CluBYOL, the pathology-adapted version of the BYOL framework, also performs better (+ 1.8 AUC

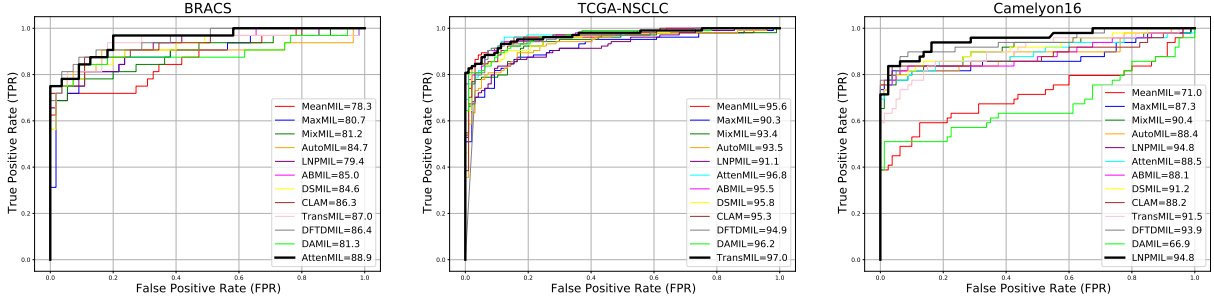


Fig. 4 ROC curves for 12 MIL methods, features extracted by ResNet18 backbone pretrained with Barlow Twins. The curves are plotted across three datasets—BRACS, TCGA-NSCLC, and Camelyon16. The best-performing method on each dataset is highlighted in a black curve with a larger line width.

points) when using pathology-specific augmentations.

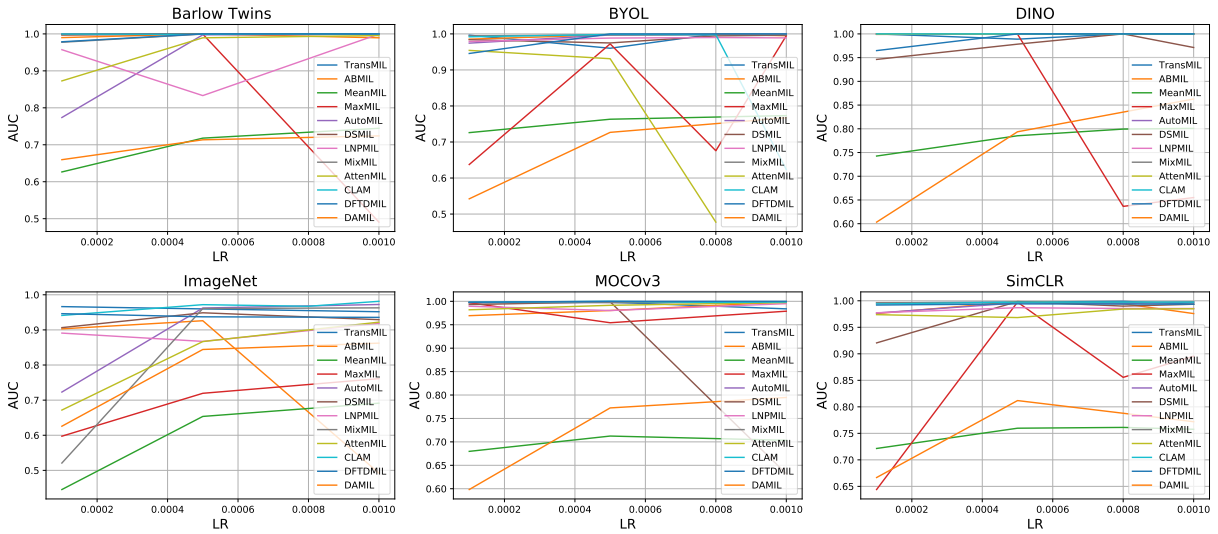


Fig. 5 Grid Search of Learning Rate on Validation set of Camelyon16 dataset with 6 pre-training methods and 12 MILs with ResNet50 backbone

Influence of ImageNet initialization. Table 6 presents the influence of the ImageNet initialization on the performance of three SSL techniques (Barlow Twins, SimCLR and DINO) combined with all MIL methods. We use as backbone the ResNet18 model. The *-IN* suffix indicates an ImageNet initialization, while no suffix means that the model has been randomly initialized. We observe that, on average, the results do not indicate a significant impact from the ImageNet initialization (+0.1, -0.1 and -0.4 AUC points, respectively). This could be explained by the fact that training for 200 epochs is sufficient to effectively learn the representations of the different tissue types in the dataset, independently from the initialization. To evaluate the robustness of our results, we compute three different trainings per SSL/MIL combination and present both the average and standard deviation. Besides the random initialization of the SSL methods, the other important sources of variability come from the random initialization of the MIL methods and the random order of the batches. Standard deviations are generally low, but they can go up to 2 AUC points. Nonetheless, even considering the estimated standard deviations, our conclusions and findings do not vary.

Qualitative results. In Fig. 6, we present qualitative results from three different slides, two correctly classified and one wrongly classified by all MIL methods. On the left, we show how MoCoV3 (Vit-Small) produces better embeddings than ImageNet pre-training, which can be leveraged by simple instance-based MIL methods to classify the slides. Indeed, on the right, we can see that the patch scores well delineate the tumor area. However, this is not the case for the attention scores of DSMIL, which can not identify as important the entire tumor area, and are thus less interpretable than the patch scores. In Fig. 12, we present more qualitative results using all MIL models and MoCoV3 (Vit-Small) as SSL method.

Similar conclusions as for Fig. 6 can be drawn when comparing instance-based and embedding-based MIL methods.

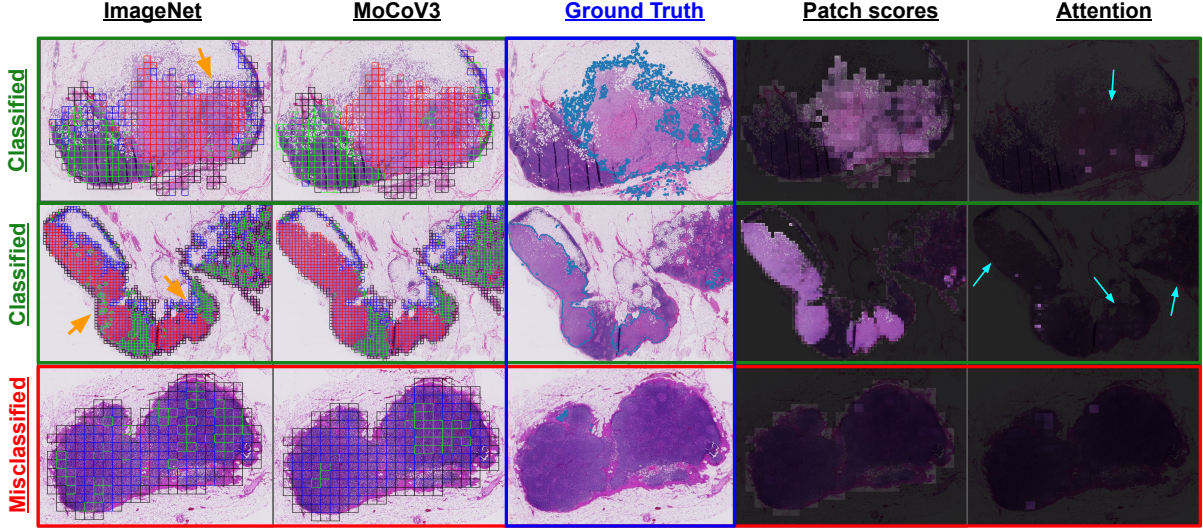


Fig. 6 *Middle*: the 3rd column presents the ground truth manual segmentation of the tumor areas from three samples of the Camelyon16 dataset. *Left*: first two columns show a K-means clustering of the patch embeddings obtained with an ImageNet (1st column) and a MoCoV3 (2nd column) pre-training using ViT-Small as backbone. Orange arrows highlight the areas correctly clustered by MoCoV3 as homogeneous tumor areas but not by imageNet. *Right*: the last two columns present the patch scores (*i.e.*, $h(f(x))$) used in MaxMIL (max score is highlighted with a red square) and the Attention scores (*i.e.*, $U(f_k, f_m)$) of DSMIL, respectively. Blue arrows indicate the areas that have been correctly identified as important by the instance-based MIL and not by the embedding-based one. The first two rows present correctly classified examples while the last row shows a sample that has been misclassified by all MIL methods. Both the patch and attention scores do not highlight the tumor area, which is also not identified in the two clustering of the embeddings. This is probably due to the fact that the tumor area is very small, with subtle changes. Future, well adapted SSL methods should be conceived for this kind of complex pathological data.

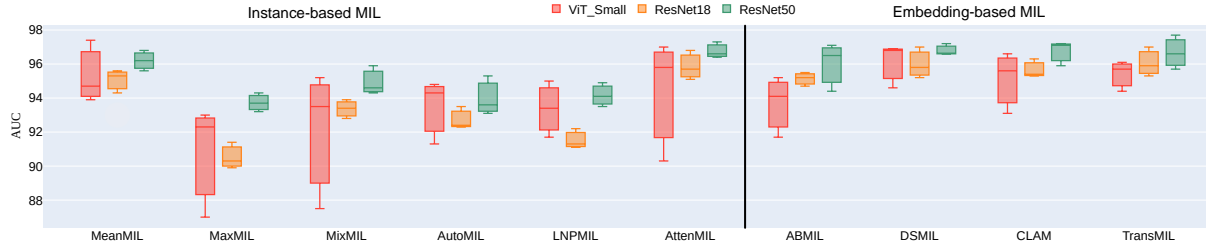


Fig. 7 The Y-axis represents the AUC scores of different MIL methods for three backbones (ViT_Small, ResNet18 and ResNet50) on the **TCGA-NSCLC** dataset at a resolution 10x. For each backbone and MIL method, we show the box-plot of 3 self-supervised pre-trainings: DINO, MOCO-V3 and Barlow Twins.

4.2 TCGA-NSCLC

Table 7 and Fig. 7 present classification results on the TCGA-NSCLC dataset. The performance of instance-based MILs is on par with embedding-based methods across all backbones. The best performing combinations of SSL and MIL are evenly spread across instance-based and embedding-based MILs. Note that embedding-based methods have 100 to 1000 times more parameters.

4.3 BRACS

Table 8 and Fig. 8 present results for the multi-class classification task on the BRACS dataset. Reported AUC scores are averages over the 3 class AUC scores. Instance and embedding-based MIL methods reach

similar AUCs, both when looking at the best combinations of MIL and SSL, and on average over all SSL methods. Furthermore, the AttenMIL method with the ResNet18 backbone pretrained with DINO achieves a new SOTA result with a 89.4 AUC score. Finally, the lower overall scores on this dataset, with respect to previous datasets, can be related to the limited number of Atypical Tumor cases.

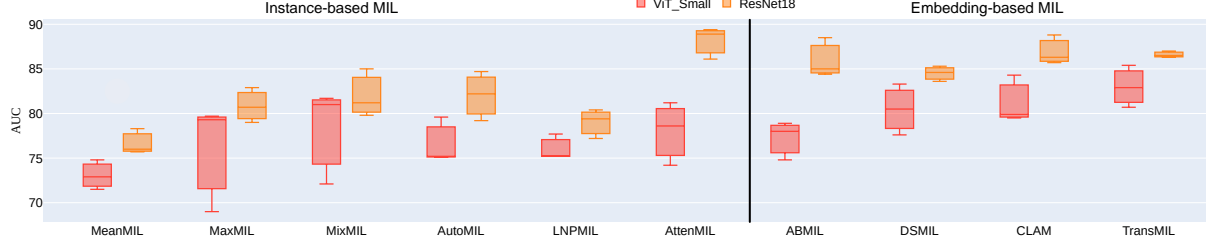


Fig. 8 The Y-axis represents the AUC scores of different MIL methods for two backbones (ViT_Small and ResNet18) on the **BRACS** dataset at a resolution 10x. For each backbone and MIL method, we show the box-plot of 3 self-supervised pre-trainings: DINO, MOCO-V3 and Barlow Twins.

4.4 VisioMel

In Table 9 and Fig. 9, we show the results on the VisioMel dataset. This is the most challenging clinical task, as indicated by the lowest overall AUC scores. Nevertheless, even here instance-based and embedding-based MIL methods perform similarly both in terms of best and averaged AUC scores.

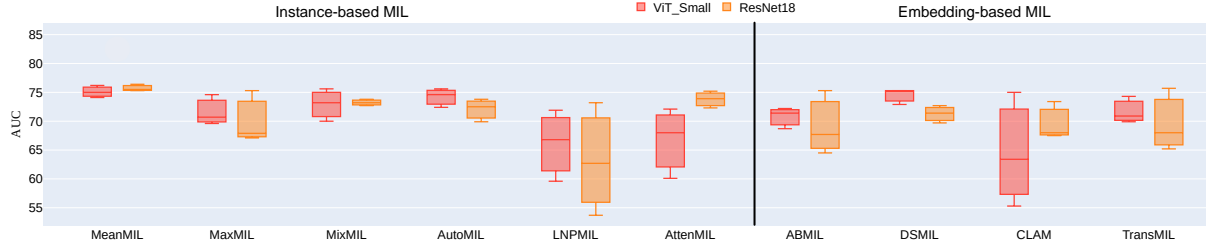


Fig. 9 The Y-axis represents the AUC scores of different MIL methods for two backbones (ViT_Small and ResNet18) on the **VisioMel** dataset at a resolution 10x. For each backbone and MIL method, we show the box-plot of 3 self-supervised pre-trainings: DINO, MOCO-V3 and Barlow Twins.

4.5 Number of Epochs for Self-Supervised Pre-Training

In Fig. 10, we illustrate the impact of the number of epochs of the SSL method on the performance of a MIL model. The x-axis denotes the number of epochs for the SSL pre-training, while the y-axis represents the average AUC scores of all MIL methods. Here, we show the results using SimCLR for ResNet18 and DINO for ViT_Small on the Camelyon16 dataset. In both examples, we show the AUC score obtained with the DSMIL method. Based on these experiments, we chose to train all SSL methods for 200 epochs, as extended pre-training enhances final performance and stabilizes the backbones, particularly for ViT ones.

4.6 Patch size

In Fig. 11, we study the impact of the patch size used during SSL pre-training on the performance of a MIL method. We tested the two most used sizes in the literature: 224x224 and 256x256. As backbone, we used ResNet18 and ViT_Small pre-trained (or not) on ImageNet ('-IN' means pre-trained on ImageNet). We show the results using SimCLR for ResNet18 and DINO for ViT_Small on the Camelyon16 dataset. For both patch sizes, we show the AUC score obtained with the DSMIL method.

We can notice that results are comparable using the two patch sizes. For computational reasons, we chose to train all SSL methods using a patch size of 256x256 (i.e., less patches and thus less computational time).

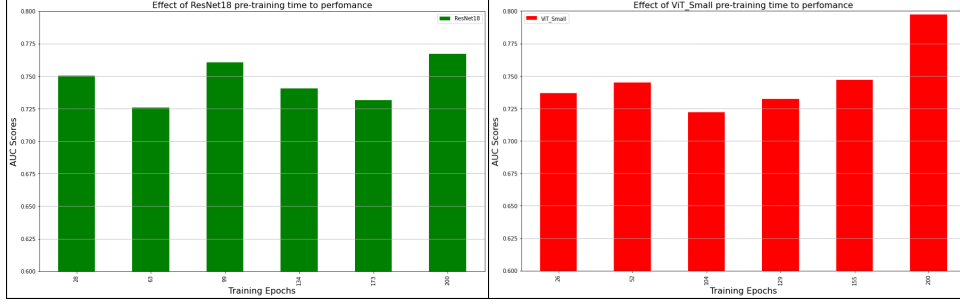


Fig. 10 Effect of the number of epochs of SSL pre-training on the performance of a DSMIL model. On the left we use SimCLR with ResNet18 and on the right DINO with ViT_Small.

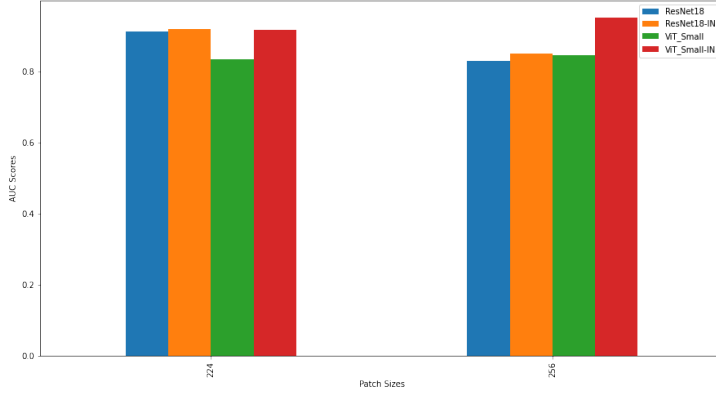


Fig. 11 Effect of the patch size used in the SSL pre-training on the performance of a DSMIL method. We used SimCLR with ResNet18 and DINO with ViT_Small. 'IN' means pre-trained on ImageNet

4.7 Discussion

In this section, we discuss the results described in the previous section. We highlight various important questions that we believe are particularly relevant to the histopathology community and provide detailed answers based on the obtained results and analyses.

Do we need complex embedding-based MIL? No. Based on our results (see Fig. 4), we can notice that instance-based MIL methods, coupled with good SSL feature extractors, are either the best or the second best performing methods, both in terms of combinations and averaged AUC scores, across all four datasets.

What is the best backbone? CNNs seem more robust than VITs. Furthermore, the performance improves with larger backbones: the best AUC with ResNet50 on Camelyon16 is 3 AUC points above the best AUC with ResNet18. The same trend is observed for TCGA-NSCLC. However, the answer also depends on the computational capability and the magnification level of the data (and thus number of samples). With great computational capability, one should use data at 20x and big architectures, such as ResNet50 or ViT.Large. However, with limited computational power or data at only 10x, one should use ResNet18.

What is the best SSL? Our best results are obtained with SSL pre-training, rather than ImageNet initialization. Among SSL methods, results may vary and the best results are obtained using either MOCOv3, DINO or Barlow Twins, with the latter being particularly advantageous due to its simple architecture and ease of tuning. We can notice that SimCLR, as already shown in the literature, falls short in terms of performance.

Do we benefit from pathology-specific SSL techniques? Pathology-adapted techniques improve the performance of traditional SSL methods. Using specific augmentations and SSL methods for pathology helps achieve better results. Since SSL methods are very sensitive to augmentations, using domain knowledge to adapt these augmentations greatly enhances generalization. For example, CluBYOL, the pathology-adapted version of the BYOL framework, significantly outperforms it by 8.6 AUC points, averaged across all MIL methods, on the Camelyon16 dataset at 10x magnification. Furthermore, CluBYOLpath achieves an even greater enhancement, adding an additional 1.8 AUC points, resulting in a total improvement of 10.4 AUC points (see Table 5).

Can we leverage foundation models? Yes, we can use foundation models as feature extractors, but

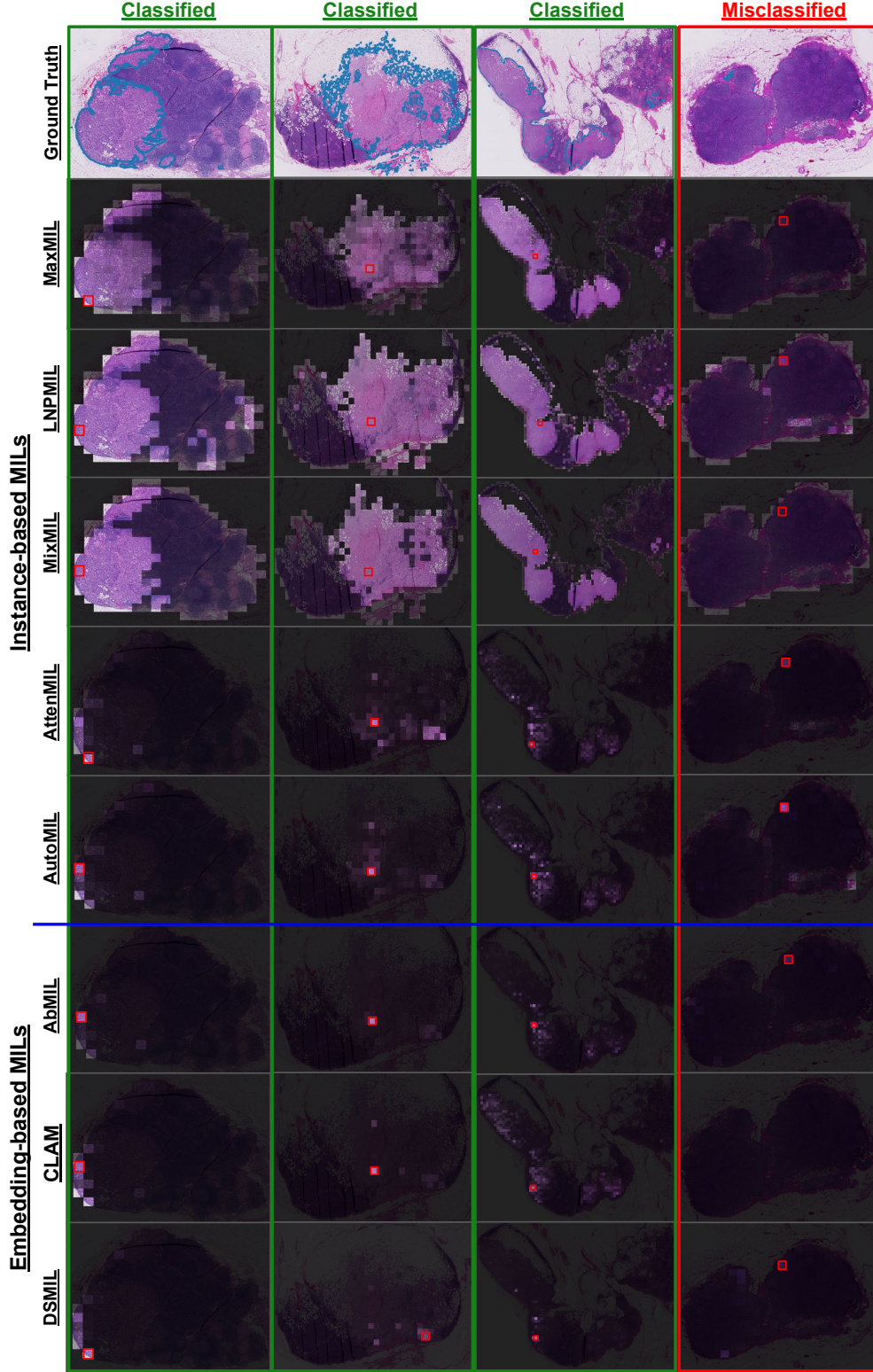


Fig. 12 Qualitative results for the Camelyon16 dataset. Each column is one case from the dataset, green text means correct classification by all MIL models, while red indicates misclassification by all MIL models. Rows represent different MIL methods, with instance-based MILs above the blue line and embedding-based MILs below it. The red squares highlight patches with the highest scores, indicating that they have the maximum contribution to the final prediction. For MaxMIL, LNPMIL, and MixMIL scores are predictions of the instance-level classifier ($h(f(x_i))$ in section 3). For the AutoMIL, scores are based on $\frac{\exp(\alpha \cdot y_k)}{\sum_{j=1}^K \exp(\alpha \cdot y_j)}$. For the remaining MIL methods, we used their attention scores.

Table 7 TCGA-NSCLC dataset at 10x. First and second-best AUC scores are in bold and underlined respectively. The best average AUC score (per MIL) is in red. ImageNet-based results are not averaged. A Welch T-Test was applied to compare the best performing method (Ref.) with the others, where (*) indicates statistically significant differences ($p < 0.05$) and "N/S" denotes not significance. The *DAMIL* method introduces additional complexity because the K-Means algorithm is applied independently on each slide.

| Arch. | Pretrain. | MeanMIL | MaxMIL | MixMIL | AutoMIL | LNPML | AttenMIL | ABMIL | DSMIL | CLAM | TransMIL | DTFDMIL | DAMIL | Avg. |
|-----------|----------------|-----------------|----------|----------|----------|----------|-----------------|----------|-----------------|-----------------|-----------------|-----------------|----------|----------|
| | # Params. | (385) | (385) | (386) | (386) | (386) | (770) | (0.05M) | (0.2M) | (0.33M) | (2.34M) | (0.23M) | (0.03M) | (-) |
| ViT-Small | ImageNet | 89.2 | 80.7 | 84.8 | 83.0 | 86.3 | 91.2 | 88.3 | 87.7 | 89.2 | 89.9 | 88.9 | 91.8 | 87.6±3.2 |
| | Barlow T. | 93.9 | 86.0 | 87.5 | 91.3 | 93.4 | 90.3 | 91.7 | 94.6 | 93.1 | 94.4 | 92.3 | 91.1 | 91.6±2.6 |
| | MOCov3 | 94.7 | 92.3 | 93.5 | 94.3 | 95.0 | 95.8 | 95.2 | 96.9 | 95.6 | 95.7 | 95.8 | 96.7 | 95.1±1.2 |
| | SimCLR | 71.2 | 65.3 | 64.4 | 72.8 | 67.5 | 64.0 | 69.2 | 76.2 | 65.6 | 61.7 | 67.2 | 70.1 | 67.9±4.0 |
| | BYOL | 79.9 | 69.0 | 68.4 | 71.7 | 70.2 | 67.4 | 73.2 | 76.4 | 72.4 | 68.5 | 72.4 | 71.9 | 71.8±3.4 |
| | DINO | 97.4 | 93.0 | 95.2 | 94.8 | 91.7 | <u>97.0</u> | 94.1 | 96.8 | 96.6 | 96.1 | 97.4 | 96.8 | 95.6±1.8 |
| | MAE | 81.3 | 79.3 | 77.6 | 80.6 | 83.3 | 85.4 | 83.7 | 90.7 | 93.6 | 91.4 | 87.7 | 89.1 | 85.3±5.0 |
| | Avg. | 86.4±9.5 | 80.8±11. | 81.1±12. | 84.2±9.7 | 83.5±11. | 83.3±13. | 84.5±10. | 88.6±8.9 | 86.1±12. | 84.6±14. | 85.5±12. | 86.0±11. | 84.6±11. |
| | <i>p-value</i> | N/S | N/S | N/S | N/S | N/S | N/S | N/S | Ref. | N/S | N/S | N/S | N/S | - |
| | # Params. | (513) | (513) | (514) | (514) | (514) | (1026) | (0.07M) | (0.33M) | (0.4M) | (2.41M) | (0.26M) | (0.04M) | (-) |
| ResNet18 | ImageNet | 88.4 | 70.4 | 83.0 | 82.7 | 82.6 | 86.1 | 88.1 | 87.7 | 87.1 | 89.9 | 88.5 | 88.7 | 85.3±5.1 |
| | Barlow T. | 95.6 | 90.3 | 93.4 | 93.5 | 91.1 | 96.8 | 95.5 | 95.8 | 95.3 | <u>97.0</u> | 94.9 | 96.2 | 94.6±2.0 |
| | MOCov3 | 95.3 | 91.4 | 93.9 | 92.3 | 92.2 | 95.1 | 94.7 | <u>97.0</u> | 96.3 | 95.9 | 97.6 | 95.7 | 94.8±1.9 |
| | SimCLR | 95.5 | 92.8 | 95.0 | 94.3 | 92.2 | 95.6 | 95.9 | 93.8 | 96.2 | 96.6 | 96.3 | 96.4 | 95.0±1.4 |
| | BYOL | 94.8 | 91.1 | 91.6 | 91.2 | 91.8 | 95.9 | 93.7 | 94.6 | 96.0 | 95.9 | 95.0 | 95.7 | 93.9±1.9 |
| | DINO | 94.3 | 89.9 | 92.8 | 92.4 | 91.3 | 95.7 | 95.2 | 95.2 | 95.4 | 95.3 | 96.5 | 95.9 | 94.2±2.0 |
| | Avg. | 95.1±0.5 | 91.1±1.0 | 93.3±1.1 | 92.7±1.1 | 91.7±0.5 | 95.8±0.6 | 95.0±0.8 | 95.3±1.1 | 95.8±0.4 | 96.1±0.6 | 96.0±1.0 | 95.9±0.3 | 94.5±1.9 |
| | <i>p-value</i> | * | * | * | * | * | N/S | * | N/S | N/S | Ref. | N/S | N/S | - |
| | # Params. | (2049) | (2049) | (2050) | (2050) | (2050) | (4098) | (0.26M) | (4.46M) | (1.18M) | (3.2M) | (0.66M) | (0.14M) | (-) |
| | ImageNet | 88.4 | 70.4 | 83.0 | 82.7 | 82.6 | 86.1 | 88.1 | 87.7 | 87.1 | 89.9 | 88.5 | 88.7 | 85.3±5.1 |
| ResNet50 | Barlow T. | 96.2 | 93.7 | 94.6 | 93.6 | 94.9 | 96.6 | 94.4 | 96.6 | 95.9 | 96.6 | 96.1 | 96.3 | 95.5±1.1 |
| | MOCov3 | 96.8 | 93.2 | 94.3 | 95.3 | 94.1 | 96.4 | 96.5 | 97.2 | 97.2 | 95.7 | 95.9 | 96.6 | 95.8±1.2 |
| | DINO | 95.6 | 94.3 | 95.9 | 93.1 | 93.5 | <u>97.3</u> | 97.1 | 96.6 | 97.1 | 97.7 | 97.1 | 96.7 | 96.0±1.5 |
| | Avg. | 96.2±0.5 | 93.7±0.4 | 94.9±0.7 | 94.0±0.9 | 94.2±0.6 | 96.8±0.4 | 96.0±1.2 | 96.8±0.3 | 96.7±0.6 | 96.7±0.8 | 96.4±0.5 | 96.5±0.2 | 95.7±1.3 |
| | <i>p-value</i> | N/S | * | * | * | * | Ref. | N/S | N/S | N/S | N/S | N/S | N/S | - |

Table 8 Multi-class BRACS dataset at 10x. The first and second-best AUC scores are in bold and underlined respectively. The best average AUC score (per MIL) is in red. ImageNet-based results are not averaged. A Welch T-Test was applied to compare the best performing method (Ref.) with the others, where (*) indicates statistically significant differences ($p < 0.05$) and "N/S" denotes not significance. The *DAMIL* method introduces additional complexity because the K-Means algorithm is applied independently on each slide.

| Arch. | Pretrain. | MeanMIL | MaxMIL | MixMIL | AutoMIL | LNPML | AttenMIL | ABMIL | DSMIL | CLAM | TransMIL | DTFDMIL | DAMIL | Avg. |
|-----------|----------------|----------|-------------|----------|----------|----------|-----------------|----------|----------|-----------------|-----------------|-----------|----------|-----------|
| | Params. | (1155) | (1155) | (1156) | (1156) | (1156) | (1540) | (0.05M) | (0.33M) | (0.2M) | (2.34M) | (0.23M) | (0.03M) | (-) |
| ViT-Small | ImageNet | 68.9 | 60.0 | 66.8 | 73.4 | 71.7 | 80.0 | 75.9 | 72.8 | 75.6 | 80.9 | 75.5 | 63.0 | 72.0±6.1 |
| | Barlow T. | 72.9 | 69.0 | 72.1 | 75.1 | 75.2 | 78.6 | 74.8 | 77.6 | 79.9 | 80.7 | 84.4 | 85.2 | 77.1±4.7 |
| | MOCov3 | 71.5 | 79.7 | 81.7 | 75.2 | 75.2 | 74.2 | 78.0 | 83.3 | 79.5 | 85.4 | 84.5 | 80.4 | 79.0±4.2 |
| | SimCLR | 50.0 | 49.2 | 52.0 | 55.8 | 58.6 | 50.5 | 54.4 | 57.1 | 60.2 | 68.9 | 84.7 | 82.1 | 60.3±11.6 |
| | BYOL | 71.3 | 85.4 | 81.7 | 76.7 | 81.0 | 75.5 | 84.9 | 84.2 | <u>85.0</u> | 80.3 | 49.2 | 55.8 | 75.9±11.3 |
| | DINO | 74.8 | 79.3 | 81.0 | 79.6 | 77.7 | 81.2 | 78.9 | 80.5 | 84.3 | 82.9 | 77.4 | 82.0 | 80.0±2.5 |
| | MAE | 64.0 | 72.9 | 75.0 | 67.2 | 64.1 | 71.7 | 71.6 | 79.0 | 74.6 | 82.5 | 79.2 | 71.0 | 72.7±5.6 |
| | Avg. | 67.4±8.5 | 72.6±11.7 | 73.9±11. | 71.6±8.0 | 72.0±7.9 | 72.0±10. | 73.8±9.6 | 77.0±9.2 | 77.2±8.4 | 80.1±5.3 | 76.6±12.6 | 76.1±10. | 74.2±10. |
| | <i>p-value</i> | * | N/S | N/S | N/S | N/S | N/S | N/S | N/S | N/S | Ref. | N/S | N/S | - |
| | Params. | (1539) | (1539) | (1540) | (1540) | (1540) | (2052) | (0.07M) | (0.33M) | (0.4M) | (2.41M) | (0.26M) | (0.04M) | (-) |
| ResNet18 | ImageNet | 71.6 | 70.3 | 60.0 | 68.8 | 69.2 | 77.5 | 78.0 | 77.3 | 76.4 | 76.1 | 80.5 | 72.8 | 73.2±5.4 |
| | Barlow T. | 78.3 | 80.7 | 81.2 | 84.7 | 79.4 | 88.9 | 85.0 | 84.6 | 86.3 | 87.0 | 86.4 | 81.3 | 83.6±3.2 |
| | MOCov3 | 76.0 | 82.9 | 85.0 | 82.2 | 77.2 | 86.1 | 84.4 | 85.3 | 85.7 | <u>89.1</u> | 86.3 | 80.6 | 83.4±3.7 |
| | SimCLR | 78.7 | 76.7 | 80.3 | 86.3 | 80.7 | 85.7 | 82.0 | 82.3 | 88.0 | 84.8 | 78.7 | 82.6 | 82.2±3.3 |
| | BYOL | 77.1 | 79.5 | 84.5 | 79.7 | 76.1 | 86.4 | 85.4 | 85.3 | 86.3 | 83.8 | 84.7 | 84.6 | 82.8±3.5 |
| | DINO | 75.7 | 79.0 | 79.8 | 79.2 | 80.4 | 89.4 | 88.5 | 83.6 | 88.8 | 86.5 | 88.2 | 79.9 | 83.2±4.6 |
| | Avg. | 77.2±1.2 | 79.8±2.0 | 82.2±2.2 | 82.4±2.8 | 78.8±1.8 | 87.3±1.5 | 85.1±2.1 | 84.2±1.1 | 87.0±1.2 | 85.7±1.2 | 85.4±3.7 | 81.8±1.7 | 83.1±3.7 |
| | <i>p-value</i> | * | * | * | * | * | Ref. | N/S | * | N/S | N/S | N/S | * | - |

it's important to use those specifically adapted to pathology. As highlighted in the previous section, there is a significant performance gap between DINOv2 and the other models, underscoring the critical role of in-domain knowledge. Unlike the other models, DINOv2 was not trained on pathological images, which emphasizes how pre-training on relevant, domain-specific data can greatly enhance performance. Among the foundation models, UNI achieves the highest performance, likely due to several factors. It benefits from having the largest and most diverse dataset, consisting of over 100 million patches, which is 5-6 times larger than others. Additionally, UNI uses the latest SSL framework and larger backbone architecture.

Should we simply use MaxMIL? Even if MaxMIL can give the best AUC score (see Table 8), it should not always be chosen. As shown in Fig. 4, the method does not always achieve the best results. In particular, we can notice that in Table 7 and 9, the best results are obtained using MeanMIL. This is because in TCGA-NSCLC and VisioMel tumor sizes are larger, typically covering more than 80% of tissue regions. Thus, averaging scores across several patches improves the result. Nevertheless, when

Table 9 VisioMel dataset at 10x. The first and second-best AUC scores are in bold and underlined respectively. The best average AUC score (per MIL) is in red. ImageNet-based results are not averaged. A Welch T-Test was applied to compare the best performing method (Ref.) with the others, where (*) indicates statistically significant differences ($p < 0.05$) and "N/S" denotes not significance. The *DAMIL* method introduces additional complexity because the K-Means algorithm is applied independently on each slide.

| Arch. | Pretrain. | MeanMIL | MaxMIL | MixMIL | AutoMIL | LNPMIL | AttenMIL | ABMIL | DSMIL | CLAM | TransMIL | DTFDMIL | DAMIL | Avg. |
|-----------|-----------|-----------------|----------|-----------------|----------|----------|----------|----------|-----------------|----------|----------|-------------|----------|----------|
| ViT-Small | Params. | (385) | (385) | (386) | (386) | (386) | (770) | (0.05M) | (0.2M) | (0.33M) | (2.34M) | (0.23M) | (0.03M) | (-) |
| | ImageNet | 69.2 | 73.9 | 71.5 | 74.3 | 52.0 | 48.5 | 59.9 | 75.3 | 60.9 | 72.9 | 70.7 | 67.9 | 66.4±8.6 |
| | Barlow T. | 74.1 | 70.7 | 73.2 | 72.4 | 59.6 | 68.0 | 71.4 | 72.9 | 55.3 | 69.9 | 71.5 | 62.3 | 68.4±5.8 |
| | MOCov3 | 75.0 | 74.6 | 75.6 | 75.6 | 71.9 | 60.1 | 72.2 | 75.3 | 75.0 | 74.3 | 74.4 | 72.3 | 73.0±4.1 |
| | SimCLR | 73.7 | 71.1 | 67.4 | 73.8 | 70.6 | 61.4 | 70.8 | 75.8 | 66.5 | 72.3 | 54.7 | 73.4 | 69.3±5.8 |
| | BYOL | 71.5 | 68.1 | 73.1 | 72.2 | 68.2 | 51.3 | 70.5 | 77.5 | 65.7 | 64.4 | 57.6 | 68.6 | 67.4±6.8 |
| | DINO | <u>76.2</u> | 69.6 | 70.0 | 74.6 | 66.8 | 72.1 | 68.7 | 75.3 | 63.4 | 70.9 | 73.8 | 73.3 | 71.2±3.6 |
| | MAE | 70.1 | 69.6 | 71.9 | 71.4 | 56.5 | 47.0 | 61.6 | 70.7 | 67.2 | 60.7 | 68.1 | 65.6 | 65.0±7.1 |
| | Avg. | <u>73.4±2.1</u> | 70.6±2.0 | 71.9±2.6 | 73.3±1.5 | 65.6±5.7 | 60.0±8.7 | 69.2±3.6 | 74.6±2.2 | 65.5±5.8 | 68.7±4.7 | 66.7±7.8 | 69.2±4.2 | 69.1±6.2 |
| | p-value | N/S | * | N/S | N/S | * | * | * | Ref. | * | * | N/S | * | - |
| ResNet18 | Params. | (513) | (513) | (514) | (514) | (514) | (1026) | (0.07M) | (0.33M) | (0.4M) | (2.41M) | (0.18M) | (0.04M) | (-) |
| | ImageNet | 74.2 | 66.1 | 73.0 | 69.4 | 50.9 | 52.9 | 70.6 | 76.1 | 67.5 | 61.4 | 67.2 | 70.1 | 66.6±7.6 |
| | Barlow T. | 75.3 | 67.9 | 72.7 | 72.5 | 62.7 | 75.2 | 67.7 | 69.7 | 68.0 | 75.7 | 61.6 | 68.2 | 69.8±4.5 |
| | MOCov3 | <u>76.4</u> | 75.3 | 73.2 | 73.8 | 53.7 | 72.3 | 75.3 | 71.4 | 67.5 | 68.0 | 61.6 | 65.4 | 69.5±6.4 |
| | SimCLR | 73.9 | 69.6 | 74.3 | 72.9 | 69.2 | 75.3 | 63.9 | 68.6 | 70.0 | 65.3 | 65.7 | 73.6 | 70.2±3.7 |
| | BYOL | 73.9 | 72.0 | 71.7 | 64.5 | 73.2 | 44.3 | 71.1 | 67.4 | 68.4 | 64.6 | 78.8 | 74.2 | 68.7±8.4 |
| | DINO | 75.5 | 67.1 | 73.8 | 69.9 | 73.2 | 73.9 | 64.5 | 72.7 | 73.4 | 65.2 | 61.1 | 73.0 | 70.3±4.5 |
| | Avg. | 75.0±1.0 | 70.4±3.0 | <u>73.1±0.9</u> | 70.7±3.4 | 66.4±7.4 | 68.2±12. | 68.5±4.3 | 70.0±1.9 | 69.5±2.1 | 67.8±4.1 | 65.8±6.7 | 70.9±3.5 | 69.7±5.8 |
| | p-value | Ref. | * | * | * | N/S | N/S | * | * | * | * | * | N/S | - |

tumor size is unknown or variable and pathology markers can be subtle, one should use an in-between approach, such as the newly proposed instance-based MIL methods.

4.8 Limitations and Perspectives

The analysis presented in this paper has some limitations that we discuss hereafter:

Multiple magnification levels. We did not use multiple magnification levels due to computational complexity. Extracting patches at different magnification levels significantly increases the dataset size, approximately 4-5 times for each higher resolution level, making it very expensive for our computational resources. However, as noted by [43] and [52], this approach improves the overall model performance, probably because it mimics pathologists, who typically examine tissues at various magnifications when making diagnostic decisions.

Other type of MIL. Here, we included in our analysis 4 embedding-based MIL methods which recently achieved SOTA results. However, other MIL methods exist which leverage, for instance, multi-levels of magnification [43, 53, 68] or graph structures [14, 19], and it would be of interest to include them in our comparison.

Influence of number of training samples N. Instance-based MIL methods, in particular MaxMIL, theoretically require more WSIs to work, since only a small number of instances within each slide (1 in MaxMIL) can actually participate in the decision [17]. At the same time, embedding-based MIL methods have more parameters and thus require more training data. In the future, it would be interesting to understand the impact of N on both classes of MIL methods.

Better pathology-adapted SSL methods. The current pathology-adapted SSL methods often involve minor modifications compared to standard SSL methods originally designed for natural images, such as adjustments in augmentation techniques, positive/negative sampling, or clustering strategies. More effort should be put into developing well-adapted SSL methods that leverage not only the different magnification levels but also the prior medical knowledge of the pathologists. An SSL approach that mimics the reasoning process of a pathologist could potentially be integrated into foundation models like UNI, enhancing their applicability and performance in WSI analysis tasks.

5 Conclusions

In this paper, we conducted a large-scale study using 6 SSL with 4 backbones, 4 foundation models and 10 MIL methods on four diverse datasets, covering binary and multi-class classification tasks with increasing clinical complexity. Our results demonstrate that simple instance-based MIL methods with very few parameters, combined with robust SSL feature extractors, can achieve competitive or superior performance than complex embedding-based MIL methods across different backbones. The newly proposed instance-based MIL methods achieve new SOTA results on BRACS and Camelyon16. By sharing

our code, pre-trained models, and insights, we aim to provide valuable resources for future research in this domain.

6 Code and Data Availability

The code for implementing this pipeline is available on GitHub at https://github.com/mammadov7/wsi_classification_pipeline. In this study, we used four publicly available datasets:

- **VisioMel** - melanoma slides data, accessible at <https://www.drivendata.org/competitions/148/visiomel-melanoma/data/>
- **TCGA-NSCLC** - non-small cell lung cancer data, obtained from the TCGA-LUAD and TCGA-LUSC projects, available via https://portal.gdc.cancer.gov/analysis_page?app=Projects
- **BRACS** - breast cancer images available at <https://www.bracs.icar.cnr.it/>
- **Camelyon16** - lymph node metastasis data, accessible at <http://gigadb.org/dataset/100439>.

7 Acknowledgements

This paper has been supported by the French National Research Agency (ANR-20-THIA-0012) and by the Hi!PARIS Center on Data Analytics and Artificial Intelligence. Furthermore, this work was performed using HPC resources from GENCI-IDRIS (Grant 2023-AD011013982R1).

References

- [1] Gurcan, M.N., Boucheron, L.E., Can, A., Madabhushi, A., Rajpoot, N.M., Yener, B.: Histopathological image analysis: a review. *IEEE reviews in biomedical engineering* **2**, 147–171 (2009) <https://doi.org/10.1109/RBME.2009.2034865>
- [2] Cheplygina, V., Bruijne, M., Pluim, J.P.W.: Not-so-supervised: A survey of semi-supervised, multi-instance, and transfer learning in medical image analysis. *Medical Image Analysis* **54**, 280–296 (2019)
- [3] Srinidhi, C.L., Ciga, O., Martel, A.L.: Deep neural network models for computational histopathology: A survey. *Medical Image Analysis* **67**, 101813 (2021)
- [4] Hou, L., Samaras, D., Kurc, T.M., Gao, Y., Davis, J.E., Saltz, J.H.: Patch-Based Convolutional Neural Network for Whole Slide Tissue Image Classification. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2424–2433. IEEE, Las Vegas, NV, USA (2016)
- [5] Wei, J.W., Tafe, L.J., Linnik, Y.A., Vaickus, L.J., Tomita, N., Hassanpour, S.: Pathologist-level classification of histologic patterns on resected lung adenocarcinoma slides with deep neural networks. *Scientific Reports* **9**(1), 3358 (2019). Number: 1 Publisher: Nature Publishing Group
- [6] Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., Joulin, A.: Emerging properties in self-supervised vision transformers. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9650–9660 (2021)
- [7] He, K., Fan, H., Wu, Y., Xie, S., Girshick, R.: Momentum contrast for unsupervised visual representation learning. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9729–9738 (2020)
- [8] Zbontar, J., Jing, L., Misra, I., LeCun, Y., Deny, S.: Barlow twins: Self-supervised learning via redundancy reduction. In: *International Conference on Machine Learning*, pp. 12310–12320. PMLR, ??? (2021)
- [9] Tschuchnig, M.E., Grubmüller, P., Stangassinger, L.M., Kreutzer, C., Couillard-Després, S., Oostingh, G.J., Hittmair, A., Gadermayr, M.: Evaluation of Multi-Scale Multiple Instance Learning to Improve Thyroid Cancer Classification. In: *Eleventh International Conference on Image Processing Theory, Tools and Applications (IPTA)*, pp. 1–6 (2022)

- [10] Qu, L., Liu, S., Liu, X., Wang, M., Song, Z.: Towards label-efficient automatic diagnosis and analysis: a comprehensive survey of advanced deep learning-based weakly-supervised, semi-supervised and self-supervised techniques in histopathological image analysis. *Physics in Medicine & Biology* (2022). Publisher: IOP Publishing
- [11] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N.: An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In: *International Conference on Learning Representations (ICLR)* (2021)
- [12] Campanella, G., Hanna, M.G., Geneslaw, L., Miraflor, A., Werneck Krauss Silva, V., Busam, K.J., Brogi, E., Reuter, V.E., Klimstra, D.S., Fuchs, T.J.: Clinical-grade computational pathology using weakly supervised deep learning on whole slide images. *Nature Medicine* **25**(8), 1301–1309 (2019) <https://doi.org/10.1038/s41591-019-0508-1>
- [13] Ilse, M., Tomczak, J., Welling, M.: Attention-based Deep Multiple Instance Learning. In: *Proceedings of the 35th International Conference on Machine Learning*, pp. 2127–2136. PMLR, ??? (2018). ISSN: 2640-3498
- [14] Li, R., Yao, J., Zhu, X., Li, Y., Huang, J.: Graph CNN for Survival Analysis on Whole Slide Pathological Images. In: Frangi, A.F., Schnabel, J.A., Davatzikos, C., Alberola-López, C., Fichtinger, G. (eds.) *Medical Image Computing and Computer Assisted Intervention – MICCAI 2018. Lecture Notes in Computer Science*, pp. 174–182. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-00934-2_20
- [15] Lu, M.Y., Williamson, D.F., 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). Publisher: Nature Publishing Group UK London
- [16] Li, B., Li, Y., Eliceiri, K.W.: Dual-stream multiple instance learning network for whole slide image classification with self-supervised contrastive learning. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14318–14328 (2021)
- [17] Shao, Z., Bian, H., Chen, Y., Wang, Y., Zhang, J., Ji, X., *et al.*: Transmil: Transformer based correlated multiple instance learning for whole slide image classification. *Advances in neural information processing systems* **34**, 2136–2147 (2021)
- [18] Zhang, H., Meng, Y., Zhao, Y., Qiao, Y., Yang, X., Coupland, S.E., Zheng, Y.: DTFD-MIL: Double-tier feature distillation multiple instance learning for histopathology whole slide image classification. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18802–18812 (2022)
- [19] Guan, Y., Zhang, J., Tian, K., Yang, S., Dong, P., Xiang, J., Yang, W., Huang, J., Zhang, Y., Han, X.: Node-aligned graph convolutional network for whole-slide image representation and classification. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18813–18823 (2022)
- [20] Wang, Y., Li, J., Metze, F.: A comparison of five multiple instance learning pooling functions for sound event detection with weak labeling. In: *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 31–35. IEEE, ??? (2019)
- [21] Bieder, F., Sandkühler, R., Cattin, P.C.: Comparison of methods generalizing max-and average-pooling (2021)
- [22] Carbonneau, M.-A., Cheplygina, V., Granger, E., Gagnon, G.: Multiple Instance Learning: A Survey of Problem Characteristics and Applications. *Pattern Recognition* **77**, 329–353 (2018) <https://doi.org/10.1016/j.patcog.2017.10.009>
- [23] Gonthier, N., Ladjal, S., Gousseau, Y.: Multiple instance learning on deep features for weakly supervised object detection with extreme domain shifts. *Computer Vision and Image Understanding* **214**,

- [24] McFee, B., Salamon, J., Bello, J.P.: Adaptive pooling operators for weakly labeled sound event detection. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* **26**(11), 2180–2193 (2018). Publisher: IEEE
- [25] Chen, R.J., Lu, M.Y., Shaban, M., Chen, C., Chen, T.Y., Williamson, D.F.K., Mahmood, F.: Whole Slide Images are 2D Point Clouds: Context-Aware Survival Prediction Using Patch-Based Graph Convolutional Networks. In: Bruijne, M., Cattin, P.C., Cotin, S., Padoy, N., Speidel, S., Zheng, Y., Essert, C. (eds.) *Medical Image Computing and Computer Assisted Intervention – MICCAI 2021. Lecture Notes in Computer Science*, pp. 339–349. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-87237-3_33
- [26] Zhao, Y., Yang, F., Fang, Y., Liu, H., Zhou, N., Zhang, J., Sun, J., Yang, S., Menze, B., Fan, X., *et al.*: Predicting lymph node metastasis using histopathological images based on multiple instance learning with deep graph convolution. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4837–4846 (2020)
- [27] Sharma, Y., Shrivastava, A., Ehsan, L., Moskaluk, C.A., Syed, S., Brown, D.: Cluster-to-Conquer: A Framework for End-to-End Multi-Instance Learning for Whole Slide Image Classification. In: *Medical Imaging with Deep Learning* (2021)
- [28] Lin, T., Yu, Z., Hu, H., Xu, Y., Chen, C.-W.: Interventional bag multi-instance learning on whole-slide pathological images. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 19830–19839 (2023)
- [29] Matsoukas, C., Haslum, J.F., Sorkhei, M., Söderberg, M., Smith, K.: What Makes Transfer Learning Work for Medical Images: Feature Reuse & Other Factors. In: *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9215–9224. IEEE Computer Society, ??? (2022)
- [30] Raghu, M., Zhang, C., Kleinberg, J., Bengio, S.: Transfusion: Understanding Transfer Learning for Medical Imaging. In: *Advances in Neural Information Processing Systems*, vol. 32. Curran Associates, Inc., ??? (2019)
- [31] Hashimoto, N., Fukushima, D., Koga, R., Takagi, Y., Ko, K., Kohno, K., Nakaguro, M., Nakamura, S., Hontani, H., Takeuchi, I.: Multi-scale Domain-adversarial Multiple-instance CNN for Cancer Subtype Classification with Unannotated Histopathological Images. In: *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3851–3860 (2020)
- [32] Kanavati, F., Toyokawa, G., Momosaki, S., Rambeau, M., Kozuma, Y., Shoji, F., Yamazaki, K., Takeo, S., Iizuka, O., Tsuneki, M.: Weakly-supervised learning for lung carcinoma classification using deep learning. *Scientific Reports* **10**(1), 9297 (2020)
- [33] Wang, X., Yan, Y., Tang, P., Bai, X., Liu, W.: Revisiting multiple instance neural networks. *Pattern Recognition* **74**, 15–24 (2018)
- [34] Grill, J.-B., Strub, F., Altché, F., Tallec, C., Richemond, P., Buchatskaya, E., Doersch, C., Avila Pires, B., Guo, Z., Gheshlaghi Azar, M., *et al.*: Bootstrap your own latent-a new approach to self-supervised learning. *Advances in neural information processing systems* **33**, 21271–21284 (2020)
- [35] He, K., Chen, X., Xie, S., Li, Y., Dollár, P., Girshick, R.: Masked autoencoders are scalable vision learners. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16000–16009 (2022)
- [36] Chen, T., Kornblith, S., Norouzi, M., Hinton, G.: A simple framework for contrastive learning of visual representations. In: *International Conference on Machine Learning (ICML)*, pp. 1597–1607. PMLR, ??? (2020)
- [37] Dehaene, O., Camara, A., Moindrot, O., Lavergne, A., Courtiol, P.: Self-Supervision Closes the Gap Between Weak and Strong Supervision in Histology. In: *ML4H NeurIPS Workshop* (2020)

- [38] Saillard, C., Dehaene, O., Marchand, T., Moindrot, O., Kamoun, A., Schmauch, B., Jegou, S.: Self-supervised learning improves dMMR/MSI detection from histology slides across multiple cancers. In: *Proceedings of the MICCAI Workshop on Computational Pathology*, pp. 191–205. PMLR, ??? (2021)
- [39] Chen, R.J., Krishnan, R.G.: Self-Supervised Vision Transformers Learn Visual Concepts in Histopathology. In: *Learning Meaningful Representations of Life, NeurIPS* (2021)
- [40] Wang, X., Yang, S., Zhang, J., Wang, M., Zhang, J., Huang, J., Yang, W., Han, X.: TransPath: Transformer-Based Self-supervised Learning for Histopathological Image Classification. In: Bruijne, M., Cattin, P.C., Cotin, S., Padoy, N., Speidel, S., Zheng, Y., Essert, C. (eds.) *Medical Image Computing and Computer Assisted Intervention – MICCAI 2021. Lecture Notes in Computer Science*, pp. 186–195. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-87237-3_18
- [41] Chen, X., Xie, S., He, K.: An Empirical Study of Training Self-Supervised Vision Transformers. In: *ICCV* (2021)
- [42] Wang, X., Xiang, J., Zhang, J., Yang, S., Yang, Z., Wang, M.-H., Zhang, J., Yang, W., Huang, J., Han, X.: SCL-WC: Cross-Slide Contrastive Learning for Weakly-Supervised Whole-Slide Image Classification. In: Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., Oh, A. (eds.) *Advances in Neural Information Processing Systems*, vol. 35, pp. 18009–18021. Curran Associates, Inc., ??? (2022)
- [43] Chen, R.J., Chen, C., Li, Y., Chen, T.Y., Trister, A.D., Krishnan, R.G., Mahmood, F.: Scaling Vision Transformers to Gigapixel Images via Hierarchical Self-Supervised Learning. In: *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 16123–16134. IEEE, New Orleans, LA, USA (2022)
- [44] Wang, X., Yang, S., Zhang, J., Wang, M., Zhang, J., Yang, W., Huang, J., Han, X.: Transformer-based unsupervised contrastive learning for histopathological image classification. *Medical Image Analysis* **81**, 102559 (2022)
- [45] Boyd, J., Liashuha, M., Deutsch, E., Paragios, N., Christodoulidis, S., Vakalopoulou, M.: Self-Supervised Representation Learning Using Visual Field Expansion on Digital Pathology. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 639–647 (2021)
- [46] Wu, W., Gao, C., DiPalma, J., Vosoughi, S., Hassanpour, S.: Improving Representation Learning for Histopathologic Images with Cluster Constraints. In: *IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 21347–21357 (2023)
- [47] Lu, M.Y., Chen, R.J., Wang, J., Dillon, D., Mahmood, F.: Semi-Supervised Histology Classification using Deep Multiple Instance Learning and Contrastive Predictive Coding. In: *Advances in Neural Information Processing Systems* (2019)
- [48] Lu, M., Pan, Y., Nie, D., Liu, F., Shi, F., Xia, Y., Shen, D.: SMILE: Sparse-Attention based Multiple Instance Contrastive Learning for Glioma Sub-Type Classification Using Pathological Images. In: *Proceedings of the MICCAI Workshop on Computational Pathology*, pp. 159–169. PMLR, ??? (2021)
- [49] Lazard, T., Lerousseau, M., Decencière, E., Walter, T.: Giga-SSL: Self-Supervised Learning for Gigapixel Images. In: *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 4305–4314 (2023)
- [50] Tang, W., Huang, S., Zhang, X., Zhou, F., Zhang, Y., Liu, B.: Multiple instance learning framework with masked hard instance mining for whole slide image classification. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4078–4087 (2023)
- [51] Liu, K., Zhu, W., Shen, Y., Liu, S., Razavian, N., Geras, K.J., Fernandez-Granda, C.: Multiple Instance Learning via Iterative Self-Paced Supervised Contrastive Learning. In: *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3355–3365 (2023)

- [52] Kang, M., Song, H., Park, S., Yoo, D., Pereira, S.: Benchmarking Self-Supervised Learning on Diverse Pathology Datasets. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 3344–3354 (2023)
- [53] Qu, L., Yang, Z., Duan, M., Ma, Y., Wang, S., Wang, M., Song, Z.: Boosting Whole Slide Image Classification from the Perspectives of Distribution, Correlation and Magnification. In: 2023 IEEE/CVF International Conference on Computer Vision (ICCV), pp. 21406–21416. IEEE, Paris, France (2023)
- [54] Javed, S.A., Juyal, D., Padigela, H., Taylor-Weiner, A., Yu, L., Prakash, A.: Additive MIL: Intrinsically Interpretable Multiple Instance Learning for Pathology. *Advances in Neural Information Processing Systems* **35**, 20689–20702 (2022)
- [55] Lerousseau, M., Vakalopoulou, M., Deutsch, E., Paragios, N.: SparseConvMIL: Sparse Convolutional Context-Aware Multiple Instance Learning for Whole Slide Image Classification. In: Proceedings of the MICCAI Workshop on Computational Pathology, pp. 129–139. PMLR, ??? (2021). ISSN: 2640-3498
- [56] Gulcehre, C., Cho, K., Pascanu, R., Bengio, Y.: Learned-Norm Pooling for Deep Feedforward and Recurrent Neural Networks. In: Calders, T., Esposito, F., Hüllermeier, E., Meo, R. (eds.) *Machine Learning and Knowledge Discovery in Databases* vol. 8724, pp. 530–546. Springer, Berlin, Heidelberg (2014). https://doi.org/10.1007/978-3-662-44848-9_34 . Series Title: Lecture Notes in Computer Science
- [57] Lee, C.-Y., Gallagher, P.W., Tu, Z.: Generalizing Pooling Functions in Convolutional Neural Networks: Mixed, Gated, and Tree. In: Proceedings of the 19th International Conference on Artificial Intelligence and Statistics, pp. 464–472. PMLR, ??? (2016)
- [58] Brancati, N., Anniciello, A.M., Pati, P., Riccio, D., Scognamiglio, G., Jaume, G., De Pietro, G., Di Bonito, M., Foncubierta, A., Botti, G., Gabrani, M., Feroce, F., Frucci, M.: BRACS: A Dataset for BReAst Carcinoma Subtyping in H&E Histology Images. *Database: The Journal of Biological Databases and Curation* **2022**, 093 (2022)
- [59] Yao, J., Zhu, X., Jonnagaddala, J., Hawkins, N., Huang, J.: Whole slide images based cancer survival prediction using attention guided deep multiple instance learning networks. *Medical image analysis* **65**, 101789 (2020)
- [60] Shen, Y., Luo, Y., Shen, D., Ke, J.: Randstainna: Learning stain-agnostic features from histology slides by bridging stain augmentation and normalization. In: International Conference on Medical Image Computing and Computer-Assisted Intervention, pp. 212–221 (2022). Springer
- [61] Caron, M., Misra, I., Mairal, J., Goyal, P., Bojanowski, P., Joulin, A.: Unsupervised Learning of Visual Features by Contrasting Cluster Assignments. In: *Advances in Neural Information Processing Systems*, vol. 33, pp. 9912–9924. Curran Associates, Inc., ??? (2020)
- [62] Oquab, M., Darcet, T., Moutakanni, T., Vo, H.V., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., Assran, M., Ballas, N., Galuba, W., Howes, R., Huang, P.-Y., Li, S.-W., Misra, I., Rabbat, M., Sharma, V., Synnaeve, G., Xu, H., Jegou, H., Mairal, J., Labatut, P., Joulin, A., Bojanowski, P.: DINOv2: Learning Robust Visual Features without Supervision. *Transactions on Machine Learning Research* (2023)
- [63] Zhou, J., Wei, C., Wang, H., Shen, W., Xie, C., Yuille, A., Kong, T.: ibot: Image bert pre-training with online tokenizer. *International Conference on Learning Representations (ICLR)* (2022)
- [64] Chen, R.J., Ding, T., Lu, M.Y., Williamson, D.F.K., Jaume, G., Song, A.H., Chen, B., Zhang, A., Shao, D., Shaban, M., Williams, M., Oldenburg, L., Weishaupt, L.L., Wang, J.J., Vaidya, A., Le, L.P., Gerber, G., Sahai, S., Williams, W., Mahmood, F.: Towards a general-purpose foundation model for computational pathology. *Nature Medicine* (2024)
- [65] Ehteshami Bejnordi, B., Veta, M., Diest, P., Ginneken, B., Karssemeijer, N., Litjens, G., Laak, J.A.W.M., 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) <https://doi.org/10.1001/jama.2017.14585>
- [66] He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)
- [67] Costa, V.G.T.d., Fini, E., Nabi, M., Sebe, N., Ricci, E.: solo-learn: A Library of Self-supervised Methods for Visual Representation Learning. *Journal of Machine Learning Research* **23**(56), 1–6 (2022)
- [68] Deng, R., Cui, C., Remedios, L.W., Bao, S., Womick, R.M., Chiron, S., Li, J., Roland, J.T., Lau, K.S., Liu, Q., *et al.*: Cross-scale multi-instance learning for pathological image diagnosis. *Medical image analysis* **94**, 103124 (2024)