[Supplementary] Interventional Bag Multi-Instance Learning On Whole-Slide Pathological Images

1. Derivation of NWGM Approximation

We will show the derivation of Normalized Weighted Geometric Mean (NWGM) approximation used in Eq.(7) in the main paper. In a MIL problem, given the bag feature, let $g(\cdot)$ be a classifier that calculates logits for the k-way bag-level classification. The approximation moves the outer sum into softmax: $\mathbb{E}[\text{Softmax}(g(\cdot))] \approx \text{Softmax}(\mathbb{E}[g(\cdot)])$. Without loss of generality, the backdoor adjustment formula in Eq.(3) can be written as:

$$P(Y=y \mid do(X=\mathbf{x})) = \sum_{c \in C} \text{Softmax}\left(g_y(\mathbf{x} \oplus \mathbf{h})\right) P(c), \tag{A1}$$

where C denotes the confounder stratifications, g_y is the classifier logit for class y, $\mathbf{h} = h(\mathbf{x}, c)$ is the feature concatenated to $\mathbf{x}$ (see Eq.(6) in the main paper), and $P(c)$ is the prior for each confounder. Then, the NWGM of Eq. (A1) can be achieved as:

$$\sum_{c \in C} \text{Softmax}\left(g_y(\mathbf{x} \oplus \mathbf{h})\right) P(c) \tag{A2}$$

$$\approx \text{NWGM}_{c \in C}\left(\text{Softmax}\left(g_y(\mathbf{x} \oplus \mathbf{h})\right)\right) \tag{A3}$$

$$= \frac{\prod_c \left[\exp\left(g_y(\mathbf{x} \oplus \mathbf{h})\right)\right]^{P(c)}}{\sum_{i=1}^{k} \prod_c \left[\exp\left(g_i(\mathbf{x} \oplus \mathbf{h})\right)\right]^{P(c)}} \tag{A4}$$

$$= \frac{\exp\left(\sum_c g_y(\mathbf{x} \oplus \mathbf{h}) P(c)\right)}{\sum_{i=1}^{k} \exp\left(\sum_c g_i(\mathbf{x} \oplus \mathbf{h}) P(c)\right)} \tag{A5}$$

$$= \text{Softmax}\left(\mathbb{E}_c\left[g_y(\mathbf{x} \oplus \mathbf{h})\right]\right), \tag{A6}$$

where Eq. (A3) follows [1] and Eq. (A4) is the definition of NWGM. Note that we set $g(\cdot)$ be a linear classifier by default, which can be written as $g(\mathbf{x} \oplus \mathbf{h}) = \mathbf{H}_1\mathbf{x} + \mathbf{H}_2\mathbf{h}$, where $\mathbf{H}_1, \mathbf{H}_2 \in \mathbb{R}^{k \times d}$ are learnable weight, d is the dimension of $\mathbf{x}$ and $\mathbf{h}$. Then, in Eq. (A6), the term inner Softmax can be written as:

$$\sum\nolimits_c g(\mathbf{x} \oplus \mathbf{h}) P(c) = \sum\nolimits_c \left(\mathbf{H}_1\mathbf{x} + \mathbf{H}_2\mathbf{h}\right) P(c) \tag{A7}$$

$$= \mathbf{H}_1\mathbf{x} + \sum\nolimits_c \mathbf{H}_2\mathbf{h} P(c) \tag{A8}$$

$$= g\left(\mathbf{x} \oplus \sum\nolimits_c \mathbf{h} P(c)\right) \tag{A9}$$

where Eq. (A8) is because the feature of $\mathbf{x}$ is same for all confounder c, and we can discard the $\mathbb{E}$ over $\mathbf{x}$. Putting Eq. (A9) into Eq. (A6), we can get the Eq.(7) in the main paper.

2. More details about feature extractors

Generally, we adopt ResNet18 [5], ViT-small [4], and CTransPath [12] as feature extractors respectively.

ResNet-18 is a basic and widely-used CNN model in the community of WSIs. We adopt the ImageNet pre-trained model officially released by PyTorch (https://download.pytorch.org/models/resnet18-5c106cde.pth). For each instance, ResNet-18 outputs the feature of 512 dimension from the penultimate layer. **ViT-small** is a typical transformer-based model. We build upon the visual transformer architecture from [10] based on the timm library [13]. We adopt the model pre-trained with MoCo V3's manner [2]. For each instance, ViT-small outputs the class token, which is of 384 dimension. **CTransPath** is hybrid CNN-transformer feature extractor, which combines the ResNet structure and Swin Transformer blocks [7]. We adopt the model pre-trained with a semantically-relevant contrastive learning (SRCL) manner [12], where the positives include augmentation views and multiple similar ones from memory bank (measured by cosine similarity metric). For each instance, CTransPath outputs the feature of 768 dimension from average pooling layer. For ViT-small and CTransPath, they are self-supervised pre-trained on 9 datasets: UniToPatho, TissueNet, NCT-CRC-HE, Colorectal cancer, Camelyon16, TCGA-NSCLC, TCGA-RCC, MIDOG, and CRAG, containing around 15 million unlabeled patches. The pretrained ViT and CTransPath can be downloaded from https://github.com/Xiyue-Wang/TransPath.

3. More details about aggregators

We use DSMIL's code base for implementation and evaluation, and build other models based on their officially released codes.

- The offical code for ABMIL can be referred to https://github.com/AMLab-Amsterdam/

AttentionDeepMIL.

- The offical code for DSMIL can be referred to https://github.com/binli123/dsmil-wsi.
- The offical code for TransMIL can be referred to https://github.com/szc19990412/TransMIL.
- The offical code for DTFD-MIL can be referred to https://github.com/hrzhang1123/DTFD-MIL.

Following their codes, we use the Adam optimizer for ABMIL and DSMIL with the cosine decay schedule [8]. The bag feature is the attention-weighted sum of instance features. For TransMIL, Lookahead optimizer [15] is employed with a weight decay of 1e-5. The bag feature is represented by the class token. We use the Adam optimizer for DFTD-MIL with `MultiStepLR` schedule. The bag feature is generated by Tier-2. The Interventional training of stage 3 can be referred to Algorithm 1.

Algorithm 1 Pseudocode of Interventional Training

```
# Inputs:
# Confounder dictionary C with shape (K, d),
# Features of instances [b_1, ..., b_n], each
    with shape (1, d)
# Ground truth Y

# Outputs:
# Bag level prediction Y_hat, Loss L for
    optimizing network parameters

# Previous MIL training:
B = aggregator_network([b_1, ..., b_n]) # B is
    the bag feature with shape (1, d)
Y_hat = classify_head(B)
L = criterion(Y_hat, Y)

# Interventional training:
B = aggregator_network([b_1, ..., b_n]) # B is
    the bag feature with shape (1, d)
B_q = linear1(B) # Projection matrix W_1, B_q
    with shape (1, l)

C.requires_grad = False # freeze C
C_k = linear2(C) # Projection matrix W_2, C_k
    with shape (K, l)

Alpha = torch.mm(C_k, B_q.T)
Alpha = F.softmax(Alpha / sqrt(l), dim=0) #
    Normalize weighted scores
C_ave = torch.mm(Alpha.T, C) # Weighted average,
    C_ave with shape (1, d)

B = torch.cat([B, C_ave], dim=1)
Y_hat = classify_head(B)
L = criterion(Y_hat, Y)
```

4. More results about DTFD-MIL (MaxMinS)

To further verify the effectiveness of IBMIL with baseline of DTFD-MIL, we switch to "MaxMinS" as the feature distillation strategy, and provide the results in Tab. 1 and Tab. 2. As can be seen, IBMIL can bring consistent performance boost under different feature distillation strategies, which demonstrates the effectiveness of our proposed scheme.

Table 1. Results on Camelyon16 dataset.

Feature Extractor	K	Precision	Recall	Accuracy	AUC
ResNet	/	84.55	75.62	79.84	79.17
	2	87.54	81.39	84.5	85.06
	4	79.04	79.37	79.84	84.77
	8	86.16	80.74	83.72	85.11
	16	81.44	81.63	82.17	84.44
CTrans	/	96.39	94.23	95.35	95.15
	2	96.95	95.19	96.12	95.75
	4	96.48	95.50	96.12	95.95
	8	96.48	95.50	96.12	95.95
	16	96.48	95.50	96.12	96.00
ViT	/	93.84	93.24	93.80	94.66
	2	95.29	92.31	93.80	94.63
	4	95.29	92.31	93.80	94.76
	8	94.21	92.93	93.80	94.66
	16	95.29	92.31	93.80	94.53

Table 2. Results on TCGA-NSCLC dataset.

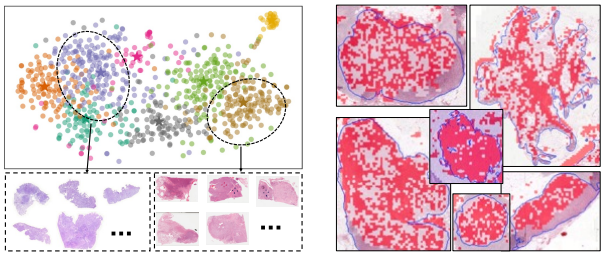
Feature Extractor	K	Precision	Recall	Accuracy	AUC
ResNet	/	88.11	88.12	88.10	92.36
	2	81.84	91.86	79.52	92.95
	4	86.13	85.71	85.71	93.41
	8	87.49	85.78	85.71	93.61
	16	90.01	89.99	90.00	94.76
CTrans	/	94.31	94.27	94.29	96.74
	2	94.31	94.27	94.29	97.71
	4	94.72	93.80	93.81	97.71
	8	94.32	94.32	94.29	97.80
	16	94.31	94.27	94.29	97.68
ViT	/	94.29	94.30	94.29	98.15
	2	94.77	94.75	94.76	98.17
	4	93.86	94.77	93.81	98.19
	8	94.77	94.75	94.76	98.22
	16	94.77	94.75	94.76	98.26

5. More Baselines

Comparison with IMIL. IMIL [6] applies instance-level physical intervention ($i.e.$, MoCo V2 style augmentation) for robust instance label prediction, while IBMIL is based on the backdoor adjustment for bag label prediction. To compare with IMIL, we apply instance-level physical intervention for bag label prediction. As shown in Tab. 3, the results ($i.e.$, ABMIL+IMIL) are even worse than baseline

Table 3. The performance with ImageNet pre-trained Res-18.

Methods	PRE	REC	ACC	AUC
ABMIL	86.71	81.71	84.50	84.07
ABMIL+IMIL	76.24	73.00	75.97	74.60
ABMIL+IBMIL	88.58	87.14	88.37	90.43
ABMIL+ColorNorm [3]	83.02	79.76	82.17	85.59



(a) (b)

Figure 1. (a) T-SNE visualization. (b) Attention maps.

on Camelyon16, since the strong augmentation achieves the causal intervention at the cost of affecting the statistical information in the bag.

Color as A confounder and Experimental Setup. In computational pathology, stain color variation is a common issue causing generalization error. From the causal lens, color is a kind of bag contextual confounders causing spurious correlations between bags and labels. Thorough evaluation was conducted on patch-based classification to highlight this issue and it claims that the conclusions generalize to WSI classification as well [9]. We conduct experiments with color normalization [3], the results ($i.e.$, ABMIL+ColorNorm) in Tab. 3 achieve better AUC than baseline. Note that IBMIL still outperforms it as there exist other confounders in general cases.

Relations to ReMix [14]. Clustering is used in ReMix and our work but with different implementations and purposes. In ReMix, clustering is performed at patch-level for each bag, and the prototypes are used to represent the bag. In our method, clustering is performed at bag-level, and the prototypes are used to approximate the confounders for backdoor adjustment.

6. Qualitative Analysis

T-SNE. In Fig. 1a, we visualize the bag features via t-SNE and denote the prototypes by stars. We empirically find that color is abstracted in some clusters. Note that confounders can be any bag contextual information ($e.g.$, color, texture or patient-specific patterns). Lacking these attribute labels hinders us from further analysis. Thus, we will turn to expert pathologist knowledge for further exploration.

Attention Map. IBMIL is proposed to empower existing bag MIL methods generally (including non-parametric ones), thus no explicit constraints are applied to attention. In Fig. 1b, the attention maps are achieved by subtraction and binarization between IBMIL and baseline, and we do find IBMIL pays more attention in tumoral regions in some cases. A potential improvement is to further incorporate attention-based interventions [11].

References

[1] Pierre Baldi and Peter Sadowski. The dropout learning algorithm. *Artificial intelligence*, 210:78–122, 2014. 1

[2] Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9640–9649, 2021. 1

[3] Cong Cong, Sidong Liu, Antonio Di Ieva, Maurice Pagnucco, Shlomo Berkovsky, and Yang Song. Colour adaptive generative networks for stain normalisation of histopathology images. *Medical Image Analysis*, 82:102580, 2022. 3

[4] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 1

[5] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016. 1

[6] Tiancheng Lin, Hongteng Xu, Canqian Yang, and Yi Xu. Interventional multi-instance learning with deconfounded instance-level prediction. *arXiv preprint arXiv:2204.09204*, 2022. 2

[7] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10012–10022, 2021. 1

[8] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016. 2

[9] David Tellez, Geert Litjens, Péter Bándi, Wouter Bulten, John-Melle Bokhorst, Francesco Ciompi, and Jeroen Van Der Laak. Quantifying the effects of data augmentation and stain color normalization in convolutional neural networks for computational pathology. *Medical image analysis*, 58:101544, 2019. 3

[10] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International Conference on Machine Learning*, pages 10347–10357. PMLR, 2021. 1

[11] Tan Wang, Chang Zhou, Qianru Sun, and Hanwang Zhang. Causal attention for unbiased visual recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3091–3100, 2021. 3

[12] Xiyue Wang, Sen Yang, Jun Zhang, Minghui Wang, Jing Zhang, Wei Yang, Junzhou Huang, and Xiao Han. Transformer-based unsupervised contrastive learning for histopathological image classification. *Medical Image Analysis*, 81:102559, 2022. 1

[13] Ross Wightman, Hugo Touvron, and Hervé Jégou. Resnet strikes back: An improved training procedure in timm. *arXiv preprint arXiv:2110.00476*, 2021. 1

[14] Jiawei Yang, Hanbo Chen, Yu Zhao, Fan Yang, Yao Zhang, Lei He, and Jianhua Yao. Remix: A general and efficient framework for multiple instance learning based whole slide image classification. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 35–45. Springer, 2022. 3

[15] Michael Zhang, James Lucas, Jimmy Ba, and Geoffrey E Hinton. Lookahead optimizer: k steps forward, 1 step back. *Advances in neural information processing systems*, 32, 2019. 2