Appendix B. Additional Method Descriptions

Appendix B.1. Deep-metric learning

The general idea of deep metric learning (DML) is to compute a representation of the inputs that preserves in the embedding space the notion of similarity present in the original space. Inputs deemed similar (in our case, of the same class) should lead to close representations in the embedding space. Several variants of DML exists that correspond to different distance measures, similarity and loss functions. For our study, we have decided to use the Normalized Softmax loss to compute the distance between feature vectors as it gave the best results after conducting preliminary experiments with various losses. The Normalized Softmax loss works by defining for each class of the dataset a proxy vector (a representative of each class). The closer a vector is from the proxy vector of its class, the smaller the loss will be. This loss therefore forces vectors of inputs of the same class to be close in the embedding space. It is defined as follows:

$$L_{Nsoftmax}(\mathbf{x}) = -\log\left(\frac{\exp(\mathbf{x}^T f(\mathbf{x})/T)}{\sum_{f(\mathbf{a})\in A}\exp(\mathbf{x}^T f(\mathbf{a})/T)}\right)$$

where A is the set of all proxies, $f(b)$ returns the proxy of the class associated to the data b, T is a hyperparameter, and x is a feature vector.

The loss is combined with a specific sampling strategy called balanced sampling [77] that constrains batches to have the same number of elements from each class of the data set. This allows to reduce the impact of class imbalance and to keep track of all the classes before updating the proxy vectors for the distance computation.

On top of those elements, the inputs are also sent through commonly used data augmentations to increase the resilience of the encoders (color jitters and random vertical/horizontal flips with probability 0.5).

Appendix B.2. Self-supervised methods

Below we describe all self-supervised methods/models that were compared in our empirical study, starting with methods trained on natural images and following with models specifically trained for histopathology.

Natural Images-based Methods

BYOL [10] is a self-supervised method targeted toward representation learning by leveraging non-contrastive learning through the use of the self-distillation principle that combines a student and a teacher network. Its training was explained in Appendix A.

Dino [8] is inspired by BYOL but introduces several improvements with respect to the latter. Dino is also based on a student and a teacher network but the composition of these networks slightly differs. For example, BYOL uses a predictor component after the encoder network that is removed in Dino. While BYOL creates two views of each image, Dino generates 8 views per images: two are global views and six are local views. Only the global views are passed through the teacher, while the student sees

all 8 views, which encourages the network to take into account both local and global details. In the version 3 of Dino [80], they have scaled up the method to be able to train models with much more parameters, the top model reaching 7B parameters, as well as increased the amount of training data, using a dataset of 1.7B images. Dinov3 also includes improvements that were made in the v2, such as the replacing of the centering/sharpening operation by the Sinkhorn-Knopp centering and the addition of a new loss objective at the patch level.

The third method, **iBOT [20],** mixes non-contrastive and generative learning. It adapts to vision the masked modeling strategy that has shown its power on language tasks. Like BYOL and Dino, iBOT creates similar image pairs by creating augmented versions of the input images and also uses two networks of similar architecture but different weights. The difference with BYOL and Dino however, is that in addition to receiving the augmented versions of the inputs, the two networks also receive augmented versions with some of their pixels masked. The representations of those 4 views (2 augmented versions and 2 masked augmented versions) are obtained by passing them through the networks, and these representations are then compared to one another by two losses. The purpose of the first loss is to push the networks to learn how to recreate the masked parts, hence the generative learning principle, while the purpose of the second one is to enforce the representation of the augmented views to be close to each other. iBOT was shown in [20] to outperform Dino on a number of tasks.

Histopathology-based Models

Phikon [22] uses iBOT as the base training strategy. Their authors wanted to see how using masked modeling techniques transfer to the digital pathology domain. The model was tested on predictive tasks at both the patch and slide levels using a set of 17 different datasets. For the patch level, these included the datasets NCT-CRC-HE-100K and Camelyon17-WILDS, which will be used in our own evaluation. Phikon is compared on those tasks with different models, including Dino and CTransPath. The authors have also tested the impact of the pre-training data, in terms of both nature and in quantity. They found that training on a larger pan-cancer dataset improves the results obtained. The first model that we used in our experiment, called Phikon, is a iBOT model fine-tuned on slides from the TCGA dataset, which was made available by the authors. We also included in our study the newest Phikon v2 [26] model, which uses Dinov2 as its training strategy, relies on a larger model (ViT-L instead of ViT-B), and was trained with more images.

cdpath21-gan [24]'s goal is to learn strong representation of tiles of whole slide images by making a model learn how to generate them (see Appendix A). cdpath21-gan is trained and evaluated on the Camelyon17 WSI dataset and the CRC patch dataset. Camelyon17 is split into patches before being used because cdpath21-gan works at the patch level. They test both the visual quality of the generated images, and the quality of the representations through classification tasks.

RetCCL [13] is a model that is specifically targeted towards retrieval. It uses a clustering-based method combined with data augmentations for its training. Similarly to usual contrastive models, RetCCL first applies data augmentations to generate 3 different views of the same input image and pass each through different networks sharing the same architecture, but with different parameters. RetCCL uses the augmented views to form positive pairs and forms negative pairs using clustering. RetCCL is centered on retrieval at the WSI level, but is also tested at the patch level. It was trained on TCGA and PAIP slides while the datasets UniToPatho and TissueNet are used for the evaluation at the patch level, and TCGA at the WSI level. In our experiments, only the feature extraction component of RetCCL is taken and used at the patch level.

CTransPath [25] is inspired by MoCo [9]. The framework is composed of 3 branches[2] : the online branch, the target branch and the shared target branch. The last two share the same structure and parameters. The backbone network selected is a modified Swin Transformer, the modification being the exchange of the original patch splitting mechanism for a CNN network. The inputs are each time an image and two views of it obtained through data augmentations. However, on top of only using those inputs at each step, CTransPath also keeps a memory bank of the previous samples. The most similar samples to the current image per cosine similarity are also used to form similar pairs. CTransPath is trained on TCGA then tested on different tasks and related datasets. In particular, it is tested on UniToPatho and TissueNet for patch retrieval, and on NCT-CRC-HE-100K for patch classification.

UNI [21] is composed of a VIT backbone trained using Dinov2's self-supervised training method. What made UNI v1 stand out at the time of its introduction is the fact that it was trained on the largest histopathology dataset to date. That dataset, called Mass-100k, contains 100M patches representing around 20 different tissue types. Most other models, on the other hand, were trained on TCGA, which contains a much more limited amount of tissue types, potentially limiting the generalization ability of these models. UNI has been tested on several kinds of tasks, such as classification, segmentation, or retrieval and has been shown to outperform all other encoders it was tested against (CTransPath, Remedis, and ResNet pretrained on ImageNet). The authors later proposed UNI v2, which relies on an even bigger training dataset as well as a bigger base model, i.e., a ViT-H of 618M parameters, as compared with the previous ViT-L of 300M parameters used in UNI v1.

H-optimus-0 [23] is a ViT model with added registers, trained using a mix of iBOT and Dinov2 methodology. The dataset on which it was trained is presented as being the most diverse histopathological dataset currently published in terms of the types of patients from whom the slides were obtained. The training dataset is a proprietary collection of 500k H&E images, such that there is never more than 3 WSI belonging to the same patient. The WSI are split into patches to be processed by the model, leading to hundreds of millions of patches. The model is designed for feature extraction, and was tested on several tasks and datasets. It reports an accuracy in classification of 0.982 for Camelyon17-WILDS and 0.962 for NCT-CRC-HE-100K. H-optimus-0 is also tested at the slide level, by combining it with the ABMIL [78] approach. Similarly to Phikon and UNI, a second version of H-optimus-0 was designed (H-Optimus-1) and it is also included in our study. The key difference between this version and the earlier ones is the amount of data used for training, which is twice that of the first dataset.

Virchow [57] is a ViT model of 632M parameters trained using Dinov2. As with previously described models, Virchow's specificity lies in the dataset on which it was trained. For its first version, this dataset is composed of more than 1.5M H&E slides, from 100k patients. It contains a wide variety of tissues (17 different types) and is mainly focused on cancerous tissues (rather than benign tissues). Virchow was tested for Pan-cancer detection and biomarker prediction. It creates embeddings at the tile level before aggregating them to perform tasks at the WSI level. The authors also tested its performance on benchmarks at the patch level to demonstrate its ability, notably on NCT-CRC-HE-100K and Camelyon17-WILDS datasets, using linear probing. They compared it to several other models including UNI and Phikon. In this paper, we use Virchowv2 [26], a newer and improved version of Virchow trained on a larger dataset containing 3.1M WSI of diverse stains (H&E and IHC).