Training state-of-the-art pathology foundation models with orders of magnitude less data

Mikhail Karasikov[1], Joost van Doorn[1], Nicolas Känzig[1], Melis Erdal Cesur[2], Hugo Mark Horlings[2], Robert Berke[1], Fei Tang[1], and Sebastian Otálora[1]

[1] Kaiko.AI, Zurich, Switzerland
[2] The Netherlands Cancer Institute, Amsterdam, The Netherlands
{mikhail,sebastian}@kaiko.ai

Abstract. The field of computational pathology has recently seen rapid advances driven by the development of modern vision foundation models (FMs), typically trained on vast collections of pathology images. Recent studies demonstrate that increasing the training data set and model size and integrating domain-specific image processing techniques can significantly enhance the model's performance on downstream tasks. Building on these insights, our work incorporates several recent modifications to the standard DINOv2 framework from the literature to optimize the training of pathology FMs. We also apply a post-training procedure for fine-tuning models on higher-resolution images to further enrich the information encoded in the embeddings. We present three novel pathology FMs trained on up to two orders of magnitude fewer WSIs than those used to train other state-of-the-art FMs while demonstrating a comparable or superior performance on downstream tasks. Even the model trained on TCGA alone (12k WSIs) outperforms most existing FMs and, on average, matches Virchow2, the second-best FM published to date. This suggests that there still remains a significant potential for further improving the models and algorithms used to train pathology FMs to take full advantage of the vast data collections.

Keywords: Foundation models · Computational pathology · Whole Slide Images.

1 Introduction

Recently, there has been an increased interest in developing vision foundation models for various types of images, including medical imaging data. These FMs generate informative representations that can be used in various downstream tasks such as classification, segmentation, object detection, etc. In particular, the development of foundation models for computational histopathology, commonly referred to in the literature as *pathology FMs*, has rapidly accelerated [19]. This progress has been primarily driven by the ever-increasing amount of unlabeled Whole Slide Image (WSI) data available in public and proprietary sources, the development of more complex model architectures (e.g., ViT [11]), and the steady refinement of training workflows (e.g., DINOv2 [20]).

Most current state-of-the-art pathology FMs are based on either the DINO [9] or DINOv2 [20] self-supervised learning (SSL) algorithms. DINOv2 was developed as an extension of DINO [9] and iBOT [31] to train general-purpose vision FMs that capture both the global context and local structure of the images, using Vision Transformers (ViTs) [11] as an underlying image encoder. Depending on the type of input images, pathology FMs can be designed to produce tile- or slide-level representations. In this work, we only consider tile-level FMs. For training a tile-level FM, the original WSIs are pre-processed to extract smaller regions, *tiles* (also often referred to in the literature as *patches*), typically of size 224×224 pixels.

One of the recent milestones in the development of pathology FMs was UNI [10], a ViT-L16 model trained with DINOv2 on over 100k WSIs from various sources, where the authors set a new standard in the performance of a pathology FM and conducted numerous experiments evaluating it on diverse downstream tasks. Very recently, the same group released a successor model UNI-2 [1] trained on over 200M tiles sampled from over 350k diverse H&E and IHC WSIs. In Kaiko-FM [2], the authors trained relatively performant models solely on TCGA and introduced *online patching*, an efficient technique for sampling WSI tiles of arbitrary size directly during training to reduce the space overhead. H-optimus-0 [22] is a ViT-g14 trained with DINOv2 on 500k WSIs with several hundreds of millions of tiles. Their model is one of the largest in terms of the number of parameters and still remains one of the best-performing published models according to various benchmarks. Another prominent model is Virchow2 [32], which is a ViT-H14 trained on a substantially larger data set comprising 3.1M WSIs. In Hibou [18], the authors trained a family of FMs with 936,441 H&E, 202,464 non-H&E, and 2,676 cytology slides sourced from 306,400 unique cases. Very recently, Atlas [3] was released as a preprint, where the authors trained a new pathology FM and demonstrated outstanding performance on the HEST [15] benchmark and six out of eight downstream tasks from *eva* [16]. However, without released weights, external evaluation of that model appears impossible.

Training pathology FMs at large scale [32,30,22,18,10,2,18,3] have pushed the frontier by amassing tens of thousands to millions of WSIs from both public and proprietary sources. However, the tendency towards scaling leaves a critical open question: is it crucial to have such a large data set in order to train a pathology FM at the state-of-the-art level, or can similar results be achieved with far fewer WSIs?

In this work, we address the question posed above and present three novel pathology FMs trained on relatively small publicly available data sets and a proprietary set of over 80k WSIs from the Netherlands Cancer Institute (NKI). Despite being trained on orders of magnitude fewer WSIs than most other state-of-the-art models published to date, our models achieve comparable and often higher performance on most downstream tasks. We additionally perform an ablation study to determine the contributions of the individual changes we made to the standard DINOv2 training workflow. Drawing an analogy be-

tween the whole range of models applied to data at different scales (from single molecules, to cells, to tissue samples, and to entire organisms) and depth zones of the ocean, we call our pathology FMs after the middle depth zone in the ocean, bathypelagic (*midnight*) zone. The shared models and the source code and data necessary to reproduce the evaluation experiments are available at `https://github.com/kaiko-ai/midnight`.

2 Methods

Training data We trained our FMs and performed the ablation study on three public collections of WSIs: TCGA, GTEx, and CPTAC, and a proprietary data set NKI-80k. TCGA contains 12k FFPE slides from 32 cancer types collected in different hospitals by the The Cancer Genome Atlas (TCGA) Research Network: `https://www.cancer.gov/tcga`. GTEx contains 25k WSIs across 23 tissue types from 838 donor individuals, collected by the Genotype-Tissue Expression project [25]. CPTAC contains 7.2k WSIs from clinical tumor samples from 13 cohorts collected by the Clinical Proteomic Tumor Analysis Consortium [12]. In addition to those three open-access data sets, we use a proprietary set of 80k WSIs (NKI-80k) from the Netherlands Cancer Institute. These slides have magnifications of 0.25 and 0.5 µm/px, most of them being at 0.25 µm/px. This data set includes mostly FFPE H&E but also Frozen Tissue and immunohistochemistry slides from 10,141 patients and 31 organs.

In our experiments, we found that including GTEx and CPTAC slides in training did not bring substantial improvements (see Results). Thus, we trained our final FMs only on the TCGA FFPE and NKI-80k slides.

Extraction of training tiles We trained our FMs on tiles of size 256×256 cropped from the original WSIs at magnifications of 2, 1, 0.5, and 0.25 µm/px. All tiles were sampled uniformly at random from arbitrary positions of the WSIs with *online patching* [2], with the foreground area threshold set to 40%. Further, to filter out low-informative tiles (e.g., those with mainly adipose tissue), we apply a filter in the HSV color space from [32]. More precisely, a tile is only accepted if ≥60% of its pixels have their hue, saturation, and value in ranges [90, 180], [8, 255], and [103, 255], respectively (see examples in Fig. A1-Left).

For all cropped tiles, we apply color augmentations in the Hematoxylin-Eosin-DAB (HED) space [24] (Fig. A1-Right). These augmentations effectively increase the diversity of the training data and help make the FM more robust to various staining methods used in the WSIs.

Self-supervised training with DINOv2 We use the DINOv2 self-distillation framework to train ViT-g14 models with 1.1B parameters (and ViT-B14 with 86M parameters in the ablation experiments) with self-supervised learning. Our algorithm is based on the original DINOv2 algorithm [20] with several modifications. First, as suggested in [32], we use a more stable KDE regularizer [28]

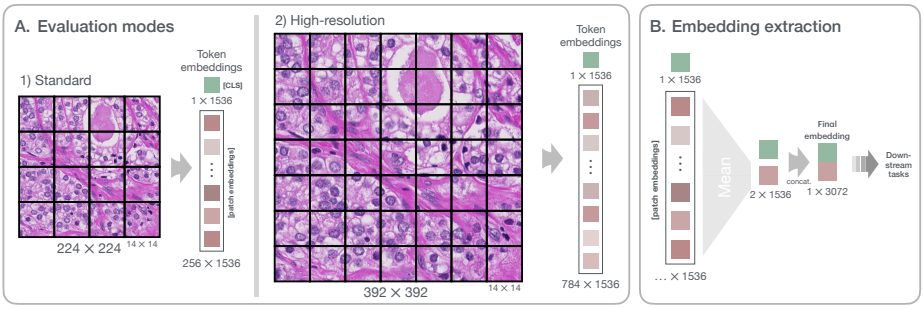

Fig. 1. Schematic representation of the FM evaluation. **Panel A:** Evaluation of a vision transformer FM at standard and high resolution. For high resolution, larger tiles of size 392×392 are cropped into $(392/14)^2 = 784$ patches of the same size 14×14 pixels. (The grids are shown schematically for simplicity. The actual numbers of patches the tiles are cropped into are 256 and 784 instead of 4^2 and 7^2, as shown in the graph.) **Panel B:** Aggregating token embeddings produced by the ViT into the final CLS+Mean token embedding.

instead of the original KoLeo loss to ensure the diversity of tile embeddings generated by the FM. We start from the checkpoints pre-trained in [20] and train on 32 Nvidia-H100 GPUs with 80 GB memory for 1M iterations with the base learning rate of 3.5×10^{-4} and the learning schedules compressed accordingly, with the batch size of 12 per GPU. (In total, it extracts throughout the training 384×10^6 tiles from the WSIs.) We accumulate gradients over every two training steps, resulting in an effective total batch size of 768.

High-resolution post-training As in [20], after training, we optionally further fine-tune the FM on larger images for 120k iterations to improve its performance, especially on high-resolution images. Similar to reducing the patch size in the underlying vision transformer studied in [6], this technique effectively increases the number of patch tokens generated from every image by the ViT. At the same time, it allows us to start from an FM pre-trained at standard resolution and thereby shorten the training on large images. More precisely, for this post-training, we increase the size of training tiles from 256 to 512 pixels, and accordingly reduce the magnification by 2-fold, to 1, 0.5, 0.25, and 0.125 µm/px, to preserve the actual size of the tile regions (512, 256, 128, and 64 µm). In addition to increasing the resolution, we also scale up the parameters of the DINO transform from 98 and 224 to 168 and 392 for the local and global crop views, respectively. Since the use of larger images increases the memory requirements, we reduce the batch size to 6 per GPU and train on 48 GPUs with accumulating gradients over every four training steps, resulting in the effective total batch size of 1152. The base learning rate in this stage is reduced to 10^{-4}. Consequently, at inference time, each input image is resized to 392×392 before passing it to the FM for the embedding generation (see Fig. 1A). Note that this resizing does

not change the actual region of the WSI contained in the tile. To the best of our knowledge, we are the first to apply this high-resolution fine-tuning procedure for training pathology FMs.

Training three new FMs We trained three new FMs: 1) We trained our first FM on the 12k TCGA WSIs alone, with the training methodology described above. We refer to this model as **Midnight-12k**. 2) For our second model, we applied the same training algorithm on both TCGA and NKI-80k combined. (Each batch was sampled from TCGA or NKI-80k at random with equal probabilities.) We refer to this model as **Midnight-92k**. 3) Finally, we fine-tuned the **Midnight-92k** FM with the high-resolution post-training technique described above, with reduced training schedules, for 120k more iterations. We refer to this model as **Midnight-92k/392**.

Evaluation methodology Our evaluation protocol is based on two open-source benchmarks: eva [16] and HEST [14]. eva includes various tile- and slide-level classification tasks and two tile segmentation tasks, assessing how well the FMs encode tissue morphology in different tissues and cancers. The original data sets from which the downstream tasks in eva were derived are summarized in Table 1. For all tasks in eva, the original tiles are resized to the desired dimensions before passing them to the FMs for embedding generation (e.g., cropping the center squares from the original 700×460 tiles at 1.995 µm/px in BreaKHis and resizing them to 224×224 results in tiles of size 224×224 at 0.97 µm/px.) For all tasks except Camelyon16 and Panda, we also disabled early-stopping in eva's default protocol to ensure that all evaluation runs fully converge.

HEST includes nine tile-level tasks that evaluate how well the FM can predict gene expression from histology images. Each task is a regression of the FM's embeddings of the 224×224 tiles at 0.5 µm/px to normalized transcript counts

Table 1. Data used in the evaluated downstream tasks. All tiles in all tasks are resized to 224×224 (or other respective dimensions) before passing to the FM for computing embeddings. (*) For slide-level tasks Camelyon16 and Panda, the values in columns 'Tile size' and 'Magnification' represent the tiles cropped from the original WSIs before resizing them to the target dimensions.

Task name	# images	Tile size	Magnification	Organ	Metric
BACH [21]	400	1536x2048	0.42 µm/px (20x)	Breast	Bal. acc.
BRACS [7]	4,539	variable	0.25 µm/px (40x)	Breast	Bal. acc.
BreaKHis [23]	7,909	700x460	1.995 µm/px (40x)	Breast	Bal. acc.
CRC-100K [17]	107,180	224x224	0.5 µm/px (20x)	Colorectal	Bal. acc.
Gleason TMA [4]	21,496	750x750	0.23 µm/px (40x)	Prostate	Bal. acc.
MHIST [29]	3,152	224x224	1.25 µm/px (8x)	Colorectal	Bal. acc.
PatchCamelyon [26]	327,680	96x96	1 µm/px (10x)	Breast	Bal. acc.
Camelyon16* [5]	399 WSIs	224x224	0.25 µm/px (40x)	Breast	Bal. acc.
Panda* [8]	1909 WSIs	448x448	0.25 µm/px (40x)	Prostate	Bal. acc.
CoNSeP [13]	41	1000x1000	0.25 µm/px (40x)	Colorectal	Dice score
MoNuSAC [27]	294	variable	0.25 µm/px (40x)	Various	Dice score
HEST (all) [15]	236,495	224x224	0.5 µm/px (20x)	Various	Pearson ρ

Table 2. Performance metrics for all evaluated FMs on the data sets from Table 1, and their average. pc10 is a tile-level classification task derived from PatchCamelyon (pc) where the training set is reduced to just ten random tiles per class (20 tiles in total). We report balanced accuracy for the classification tasks, dice score (no background) for semantic segmentation (cnsp, mnsc), and the average Pearson correlation for the nine HEST regression tasks. All classification tasks use CLS+Mean embeddings (the concatenation of the CLS token and the mean of all patch tokens in the output of the ViT). For all results, see Extended Table A1.

Model name	#WSIs	pc10	bach	brcs	bkhs	crc	glsn	mhst	pc	c16	pnd	cnsp	mnsc	HEST	Avg.
Midnight-92k/392	92k	**.900**	**.904**	**.646**	.802	.966	**.807**	.828	**.951**	.868	.651	**.662**	**.708**	.415	**.778**
UNI-2	350k	**.885**	**.924**	**.651**	**.863**	**.970**	.777	.829	**.951**	**.873**	**.666**	.626	.644	**.431**	**.776**
Midnight-92k	92k	**.882**	.889	.615	.793	**.967**	**.823**	.831	.948	**.872**	.643	.629	.656	**.425**	**.767**
Virchow2	3.1M	.835	.890	.633	.818	.966	.791	**.865**	.938	.860	.646	.640	.674	.403	.766
Midnight-12k	12k	.803	**.907**	.639	.840	**.967**	.790	.815	.931	.869	.656	.625	.664	.412	.763
Kaiko-B8	29k	.799	.876	.641	**.842**	.960	.761	.830	.920	.836	.650	**.644**	.686	.391	.757
tcga-100M	12k	.789	.873	.619	.814	**.968**	**.798**	.808	.928	**.870**	**.675**	.622	.656	.415	.757
H-Optimus-0	500k	.831	.752	.620	.813	.962	.769	**.850**	.943	.847	**.672**	**.644**	**.687**	**.425**	.755
Prov_GigaPath	171k	.853	.794	.626	**.846**	.959	.727	.831	.944	.812	.657	.628	**.688**	.405	.752
Hibou-L	1.1M	.825	.792	**.643**	.767	.954	.766	**.850**	**.949**	.852	.654	**.646**	.668	.397	.751
UNI	100k	.833	.797	.613	.808	.954	.759	.841	.937	.854	.662	.627	.662	.391	.749
Phikon	12k	.826	.744	.579	.715	.946	.743	.824	.919	.822	.648	.624	.644	.377	.724
Phikon-v2	60k	.756	.737	.607	.725	.953	.753	.796	.900	.807	.634	.626	.645	.391	.718
Lunit	36k	.763	.785	.627	.759	.943	.758	.785	.905	.759	.604	.600	.630	.362	.714
vitg14 (nat. img.)	0	.721	.724	.578	.783	.943	.740	**.855**	.881	.500	.509	.565	.614	.351	.674
vitg14 (initial)	0	.652	.474	.413	.425	.754	.459	.578	.763	.526	.304	.462	.432	.166	.493

of the top 50 highly variable genes, measured at the respective positions of the tiles. Performance in HEST is measured by the Pearson correlation coefficient between the predicted and actual gene expression, computed across all patients.

3 Results and Discussion

Reaching state-of-the-art performance with less data We evaluated the performance of our FMs and several other state-of-the-art FMs on the downstream tasks described above. For every model, we evaluated both the CLS token and the CLS+Mean token embeddings (the concatenation of the CLS token and the mean of all (image_size/patch_size)^2 patch tokens in the vision transformer, see Fig. 1B). For HEST, we only report the aggregate average of Pearson correlations. It can be seen (Table 2) that even our model **Midnight-12k** trained on just 12k WSIs is superior to most other existing FMs, and is only marginally different from Virchow2 despite being trained on 258× fewer WSIs (12k vs. 3.1M).

The **Midnight-92k** model trained on the TCGA and NKI-80k WSIs (92k WSIs in total) slightly surpasses Virchow2, and is just 0.009 behind UNI-2 (see Table 2) despite being trained on 4× fewer WSIs (92k vs. 350k). Note that UNI-2 [1] was released in Jan. 2025 as a successor of UNI [10]. Despite UNI-2 having used significantly more data for training, our models demonstrate a comparable and sometimes superior performance on the considered downstream

tasks. Unfortunately, we could not add Atlas [3] to our evaluation because we did not have the model weights for it.

Finally, our post-trained model **Midnight-92k/392** demonstrated a superior average accuracy to all other models in the benchmark and surpassed UNI-2 with an average margin of 0.002. For this evaluation, all the images were resized from their original size specified in Table 1 to 392×392, instead of resizing them to 224×224 as for all other models. The results (Table 2) indicate that the high-resolution post-training improved the base **Midnight-92k** model especially significantly on the segmentation metrics, CoNSeP and MoNuSAC. Notably, on the PCam (10 shots) task, which is derived from PatchCamelyon by reducing the training set to just ten random tiles per class (20 in total) for every evaluation run and averaging the test accuracy over 50 training runs, this model achieves the balanced accuracy of 0.90 and surpasses all other evaluated models. However, the performance on the Camelyon16 and HEST tasks has significantly degraded, which needs further investigation. Overall, this still resulted in the absolute best-performing model among all the evaluated models.

To ensure a fair comparison, we additionally evaluated UNI [10] on all downstream tasks with resized images. (In [10], they mention a fine-tuning performed with larger 512×512 images but without the details about their procedure.) However, the performance of UNI only degraded on images resized to 512×512 (see row 'UNI/512' in Extended Table A1), e.g., 0.89 on PCam, which suggests that their fine-tuning procedure was of a different nature than ours. We also evaluated UNI-2 in the same way as **Midnight-92k/392**, on 392×392 images; however, that also yielded slightly lower performance (see row 'UNI-2/392' in Extended Table A1), which again suggests that UNI-2 does not benefit from evaluating on larger images at higher resolution.

Notably, all evaluated pathology FMs surpass the baseline ViT-g14 model trained on natural images ('vitg14 (nat. img.)' in Table 2) with a large margin, which highlights the importance of developing domain-specific pathology FMs. However, all pathology FMs in our benchmark perform relatively poorly on MHIST, where the baseline model trained on natural images is the second-best model. This suggests that there still remains a potential for improvement, which sets a particular aim for future work.

Ablation experiments To measure the effect of the adaptations we made to the baseline training workflow, we performed an ablation study, where we trained several smaller ViT-B14 models (Table 3). These training runs were done for 500k iterations with 4 GPUs, a batch size of 64 per GPU, and accumulating gradients over every three training steps, resulting in an effective total batch size of 768.

The first four experiments evaluated the importance of the HSV filter, the KDE regularizer, and the HED augmentations. Without replacing the default KoLeo regularizer with KDE and without the HSV filter, our training did not converge, thus, we report 'n/a' in the first row of Table 3. After adding the HSV filter, the training converged to an average accuracy of 0.704 on eva, which was still far from 0.753, obtained with the final config. The HED color augmentations

Table 3. Performance of the models in the ablation study. Each row corresponds to a single ViT-B14 model trained with the modifications specified by check marks.

TCGA	NKI-80k	CPTAC	GTEx	HSV	KDE	HED	eva	HEST
✓							n/a	n/a
✓				✓			0.704	0.367
✓				✓	✓		0.754	0.367
✓				✓	✓	✓	0.753	0.376
✓	✓			✓	✓	✓	0.759	0.374
✓		✓		✓	✓	✓	0.744	**0.380**
✓			✓	✓	✓	✓	0.750	0.363
	✓	✓	✓	✓	✓	✓	0.742	0.368
✓		✓	✓	✓	✓	✓	0.742	0.375
✓	✓		✓	✓	✓	✓	0.750	0.368
✓	✓	✓		✓	✓	✓	0.765	0.373
✓	✓	✓	✓	✓	✓	✓	**0.768**	0.375

improved the performance on HEST but did not have a large effect on eva. However, we applied it anyway to help make the FMs more robust to different stainings.

Next, we added each of NKI-80k, CPTAC, and GTEx, to evaluate their contribution when added to TCGA. (The tiles were sampled from both data sets with equal probability.) The results (rows 5, 6, and 7 in Table 3) show that all three had a rather small impact, with NKI-80k bringing the highest average gain in accuracy. We also ran data ablations relative to the baseline run on all four data sets: TCGA, GTEx, CPTAC, and NKI-80k (the five last rows of Table 3). Here, we excluded each data set at a time and trained a ViT-B14 on the remaining three data sets. The results show that removing GTEx and CPTAC only marginally affected the FM's final performance, while removing TCGA and NKI-80k resulted in a higher loss.

Last, to check whether we could get an FM with a comparable performance to that of **Midnight-12k** even with less data, we trained the large ViT-g14 model on just 10% of the TCGA slides (∼1k WSIs). The resulting performance was far lower, e.g., only 0.9 on PCAM after 500k iterations. We also trained ViT-g14 on 100M distinct tiles randomly sampled from all the 12k TCGA slides, which resulted in a slightly lower performance than that of **Midnight-12k** (row '**tcga-100M**' in Extended Table A1).

Image segmentation Identifying different cell types can be essential not only for making an accurate diagnosis but also for understanding tumor behavior by analyzing the cellular composition of the micro-environment. In addition to systematically evaluating the FMs' capability to segment and classify cells in images on the CoNSeP and MoNuSAC tasks (evaluated in eva), here, we selected two images from the CoNSeP data set for a clear visual demonstration and performed the standard semantic segmentation procedure implemented in eva for four models: ViT-g14 (natural images), Lunit, Virchow2, and our models **Midnight-12k** and **Midnight-92k/392**.

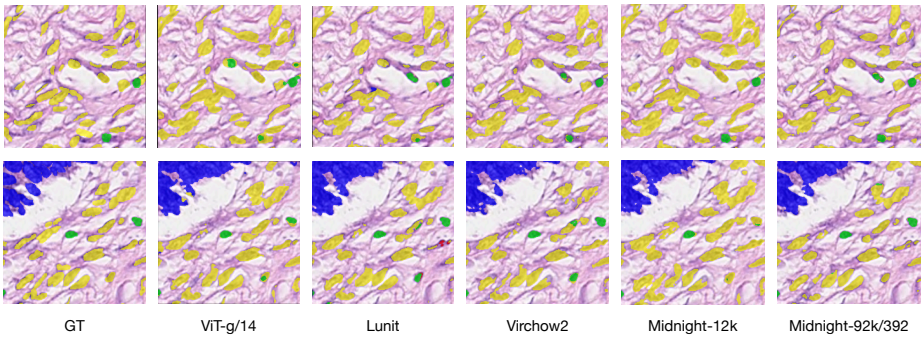

Fig. 2. Examples of segmentation performed with different FMs on two tiles from the CoNSeP data set: ViT-g14 (natural images), Lunit, Virchow2, and our models **Midnight-12k** and **Midnight-92k/392**. Ground truth is shown on the left side — green: inflammatory, blue: epithelial, yellow: spindle-shaped nuclei.

All pathology FMs produce segmentations (Fig. 2) that are noticeably better than the baseline model, which again highlights the importance of training the FMs on pathology images. Notably, our models produce segmentations that are comparable in quality to those from Virchow2, despite being trained on 34–258× fewer WSIs (12–92k vs. 3.1M).

Tile-level classification with FM for slide-level segmentation Metastasis to regional lymph nodes is an early sign of malignant spread. Thus, detecting lymph node metastasis is crucial in many cancer types, as it upstages the disease and impacts both clinical outcomes and treatment strategies. To demonstrate how tile-level pathology FMs perform on slide-level tasks, we trained tile-level

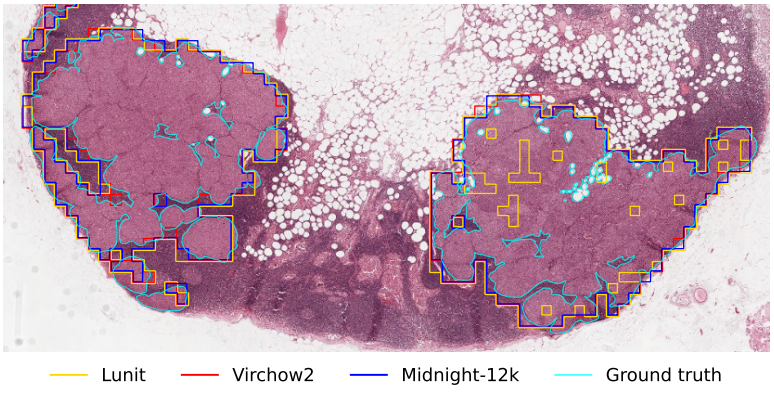

Fig. 3. Tile predictions for Lunit, Virchow2, and **Midnight-12k** and ground truth annotations for slide `test_040` from Camelyon16.

downstream classifiers to detect lymph node metastases in breast cancer on the Camelyon16 data with three different FMs: Lunit, Virchow2, and **Midnight-12k**. We applied these classifiers on a randomly picked slide `test_040` (Fig. 3). It can be seen that the predictions for **Midnight-12k** and Virchow2 are nearly identical and close to the expert annotations. At the same time, the weaker FM Lunit generates far less accurate predictions, which clearly shows the practical importance of using higher-performing FMs to get qualitatively better results on downstream tasks.

4 Conclusion

We have presented three new pathology FMs trained on up to two orders of magnitude fewer WSIs than some state-of-the-art models, yet achieving comparable or superior performance on downstream tasks. We have shown that even with a relatively basic setup, it is possible to train a high-performing pathology FM with far fewer WSIs than one may have previously thought necessary.

We make our **Midnight-12k** model trained solely on TCGA (which currently ranks third among all publicly available models) open for download from `https://huggingface.co/kaiko-ai/midnight` under the MIT license to encourage further research and reproducibility.

With our main results, we do not mean to imply that small data sets are always sufficient to reach state-of-the-art performance. On the contrary, our strong results with magnitudes fewer WSIs show that foundation model training in pathology remains far from saturation. In other words, we believe that there is still significant unrealized potential in today's algorithms — potential that can be tapped at truly large scale. As pathology AI continues to evolve, we believe our work makes a solid contribution to the collective efforts of devising better pathology FMs. This brings us another step closer to achieving real impact in clinics and lowering the burden on pathologists while ultimately improving the quality of provided healthcare.

Acknowledgements

We thank the Core Facility for Molecular Pathology and Biobanking of the Netherlands Cancer Institute for providing the NKI-80k digitized slides with approval from the Institutional review board (IRBdm22-188). We also thank Prof. Dr. Lodewyk Wessels for his support throughout this project. The GTEx data used for the analyses described in this manuscript were obtained from the GTEx Portal `https://gtexportal.org` on October 1, 2024. The TCGA slides used in this work were generated by the TCGA Research Network: `https://www.cancer.gov/tcga`. We thank Nils Eckstein, Moritz Platscher, Thomas Hufener, and others at kaiko.ai for the helpful discussions and their valuable feedback on this manuscript. We also thank Dang Nguyen, Gianmaria Genetlici, and the rest of the Infrastructure Team at kaiko.ai for maintaining and optimizing the computing resources used in this work.

References

1. MahmoodLab/UNI2-h (revision d517a8d) (2025), https://huggingface.co/MahmoodLab/UNI2-h, accessed on 11 Feb 2025
2. kaiko. ai, Aben, N., de Jong, E.D., Gatopoulos, I., Känzig, N., Karasikov, M., Lagré, A., Moser, R., van Doorn, J., Tang, F.: Towards large-scale training of pathology foundation models (2024), https://arxiv.org/abs/2404.15217
3. Alber, M., Tietz, S., Dippel, J., Milbich, T., Lesort, T., Korfiatis, P., Krügener, M., Cancer, B.P., Shah, N., Möllers, A., et al.: Atlas: A novel pathology foundation model by mayo clinic, charit\'e, and aignostics. arXiv preprint arXiv:2501.05409 (2025)
4. Arvaniti, E., et al.: Automated gleason grading of prostate cancer tissue microarrays via deep learning. Scientific reports **8**(1), 12054 (2018)
5. Bejnordi, B., et al.: Diagnostic assessment of deep learning algorithms for detection of lymph node metastases in women with breast cancer. Jama **318**(22), 2199–2210 (2017)
6. Beyer, L., Izmailov, P., Kolesnikov, A., Caron, M., Kornblith, S., Zhai, X., Minderer, M., Tschannen, M., Alabdulmohsin, I., Pavetic, F.: Flexivit: One model for all patch sizes. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 14496–14506 (2023)
7. Brancati, N., Anniciello, A.M., Pati, P., Riccio, D., Scognamiglio, G., Jaume, G., De Pietro, G., Di Bonito, M., Foncubierta, A., Botti, G., et al.: Bracs: A dataset for breast carcinoma subtyping in h&e histology images. Database **2022**, baac093 (2022)
8. Bulten, W., Kartasalo, K., Chen, P.H.C., Ström, P., Pinckaers, H., Nagpal, K., Cai, Y., Steiner, D.F., Van Boven, H., Vink, R., et al.: Artificial intelligence for diagnosis and gleason grading of prostate cancer: the panda challenge. Nature medicine **28**(1), 154–163 (2022)
9. 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)
10. Chen, R.J., Ding, T., Lu, M.Y., Williamson, D.F., Jaume, G., Chen, B., Zhang, A., Shao, D., Song, A.H., Shaban, M., et al.: Towards a general-purpose foundation model for computational pathology. Nature Medicine (2024)
11. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020)
12. Edwards, N.J., Oberti, M., Thangudu, R.R., Cai, S., McGarvey, P.B., Jacob, S., Madhavan, S., Ketchum, K.A.: The cptac data portal: a resource for cancer proteomics research. Journal of proteome research **14**(6), 2707–2713 (2015)
13. Graham, S., Vu, Q.D., Raza, S.E.A., Azam, A., Tsang, Y.W., Kwak, J.T., Rajpoot, N.: Hover-net: Simultaneous segmentation and classification of nuclei in multi-tissue histology images. Medical image analysis **58**, 101563 (2019)
14. Jaume, G., Doucet, P., Song, A.H., Lu, M.Y., Almagro-Perez, C., Wagner, S.J., Vaidya, A.J., Chen, R.J., Williamson, D.F.K., Kim, A., Mahmood, F.: Hest-1k: A dataset for spatial transcriptomics and histology image analysis. In: Advances in Neural Information Processing Systems (Dec 2024)
15. Jaume, G., Doucet, P., Song, A.H., Lu, M.Y., Almagro-Pérez, C., Wagner, S.J., Vaidya, A.J., Chen, R.J., Williamson, D.F., Kim, A., et al.: Hest-1k: A

dataset for spatial transcriptomics and histology image analysis. arXiv preprint arXiv:2406.16192 (2024)
16. kaiko.ai, Gatopoulos, I., Känzig, N., Moser, R., Otálora, S.: eva: Evaluation framework for pathology foundation models. In: Medical Imaging with Deep Learning (2024), https://openreview.net/forum?id=FNBQOPj18N
17. Kather, J.N., Halama, N., Marx, A.: 100,000 histological images of human colorectal cancer and healthy tissue (Apr 2018). https://doi.org/10.5281/zenodo.1214456, https://doi.org/10.5281/zenodo.1214456
18. Nechaev, D., Pchelnikov, A., Ivanova, E.: Hibou: A family of foundational vision transformers for pathology. arXiv preprint arXiv:2406.05074 (2024)
19. Ochi, M., Komura, D., Ishikawa, S.: Pathology foundation models. JMA journal **8**(1), 121–130 (2025)
20. Oquab, M., et al.: Dinov2: Learning robust visual features without supervision. arXiv preprint arXiv:2304.07193 (2023)
21. Polónia, A., Eloy, C., Aguiar, P.: Bach dataset : Grand challenge on breast cancer histology images (May 2019). https://doi.org/10.5281/zenodo.3632035, https://doi.org/10.5281/zenodo.3632035
22. Saillard, C., Jenatton, R., Llinares-López, F., Mariet, Z., Cahané, D., Durand, E., Vert, J.P.: H-optimus-0 (2024), https://github.com/bioptimus/releases/tree/main/models/h-optimus/v0
23. Spanhol, F.A., Oliveira, L.S., Petitjean, C., Heutte, L.: A dataset for breast cancer histopathological image classification. Ieee transactions on biomedical engineering **63**(7), 1455–1462 (2015)
24. Tellez, D., et al.: Quantifying the effects of data augmentation and stain color normalization in convolutional neural networks for computational pathology. Medical image analysis **58**, 101544 (2019)
25. The GTEx Consortium, et al.: The genotype-tissue expression (gtex) pilot analysis: Multitissue gene regulation in humans. Science **348**(6235), 648–660 (2015). https://doi.org/10.1126/science.1262110, https://www.science.org/doi/abs/10.1126/science.1262110
26. Veeling, B., Linmans, J., Cohen, T., Welling, M.: Rotation equivariant cnns for digital pathology (Sep 2018). https://doi.org/10.1007/978-3-030-00934-2_24, https://doi.org/10.1007/978-3-030-00934-2_24
27. Verma, R., Kumar, N., Patil, A., Kurian, N.C., Rane, S., Graham, S., Vu, Q.D., Zwager, M., Raza, S.E.A., Rajpoot, N., et al.: Monusac2020: A multi-organ nuclei segmentation and classification challenge. IEEE Transactions on Medical Imaging **40**(12), 3413–3423 (2021)
28. Wang, T., Isola, P.: Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In: International conference on machine learning. pp. 9929–9939. PMLR (2020)
29. Wei, J., Suriawinata, A., Ren, B., Liu, X., Lisovsky, M., Vaickus, L., Brown, C., Baker, M., Tomita, N., Torresani, L., et al.: A petri dish for histopathology image analysis. In: International Conference on Artificial Intelligence in Medicine. pp. 11–24. Springer (2021)
30. Xu, H., Usuyama, N., Bagga, J., Zhang, S., Rao, R., Naumann, T., Wong, C., Gero, Z., González, J., Gu, Y., et al.: A whole-slide foundation model for digital pathology from real-world data. Nature pp. 1–8 (2024)
31. Zhou, J., Wei, C., Wang, H., Shen, W., Xie, C., Yuille, A., Kong, T.: ibot: Image bert pre-training with online tokenizer. arXiv preprint arXiv:2111.07832 (2021)

32. Zimmermann, E., Vorontsov, E., Viret, J., Casson, A., Zelechowski, M., Shaikovski, G., Tenenholtz, N., Hall, J., Klimstra, D., Yousfi, R., et al.: Virchow2: Scaling self-supervised mixed magnification models in pathology. arXiv preprint arXiv:2408.00738 (2024)

A Appendix

A.1 Supplementary Figures

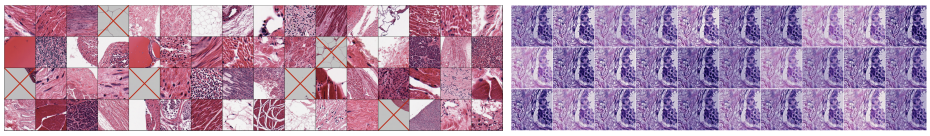

Fig. A1. Left: Random tiles cropped from TCGA FFPE slides at 0.5µm/px that passed the HSV filter and those filtered out. **Right:** Random HED augmentations applied to a single tile sampled from the BACH data set.

A.2 Extended Tables

Table A1. Performance of all evaluated FMs on the evaluated downstream tasks and their average. PCam (10 shots) is a tile-level classification task derived from PCam where the training set is reduced to just ten random tiles per class (20 tiles in total). We report balanced accuracy for the classification tasks, dice score (no background) for semantic segmentation (CoNSeP and MoNuSAC), and the average Pearson correlation for the nine HEST regression tasks. All models were evaluated in two variants on the classification tasks, using CLS+Mean token embeddings (default) and CLS only. (For the latter, the respective rows are highlighted in light grey.) The models are sorted by the average performance with the CLS+Mean tokens, and the top three values for each task are highlighted in bold. (The evaluations with the CLS tokens, rows in grey, are not taken into account for this sorting.) Each subscript for the tasks in eva represents the standard deviation of the sample mean computed over multiple runs of the respective downstream task. (*) For HEST, the subscripts represent standard deviations of the predicted gene expressions, averaged across the nine HEST tasks. (**) For the average (last column), the subscripts represent standard deviations averaged across all 13 evaluated downstream tasks.

Model name	#WSIs	PCam (10 shots)	BACH	BRACS	BreaKHis	CRC-100K	Gleason	MHIST	PCam	Cam16 (small)	Panda (small)	CoNSeP	MoNuSAC	HEST (*)	Avg. (**)
Midnight-92k/392	92k	**.900**$_{\pm.005}$	.904$_{\pm.002}$	**.646**$_{\pm.001}$	.802$_{\pm.000}$	.966$_{\pm.000}$	**.807**$_{\pm.001}$	.828$_{\pm.000}$	**.951**$_{\pm.000}$	.868$_{\pm.002}$	.651$_{\pm.002}$	**.662**$_{\pm.001}$	**.708**$_{\pm.002}$	.415$_{\pm.038}$	**.778**$_{\pm.004}$
Midnight-92k/392 [CLS]	92k	.900$_{\pm.005}$	.906$_{\pm.001}$	.642$_{\pm.001}$	.850$_{\pm.000}$	.964$_{\pm.000}$	.809$_{\pm.001}$	.825$_{\pm.000}$	.951$_{\pm.000}$	.831$_{\pm.002}$	.633$_{\pm.002}$	.663$_{\pm.001}$	.707$_{\pm.002}$	.384$_{\pm.044}$	.774$_{\pm.005}$
UNI-2	350k	**.885**$_{\pm.005}$	**.924**$_{\pm.002}$	**.651**$_{\pm.004}$	**.863**$_{\pm.002}$	**.970**$_{\pm.000}$	.777$_{\pm.003}$	.829$_{\pm.001}$	**.951**$_{\pm.000}$	**.873**$_{\pm.002}$	**.666**$_{\pm.003}$	.626$_{\pm.000}$	.644$_{\pm.002}$	**.431**$_{\pm.039}$	**.776**$_{\pm.005}$
UNI-2 [CLS]	350k	.887$_{\pm.005}$	.914$_{\pm.002}$	.661$_{\pm.002}$	.860$_{\pm.001}$	.965$_{\pm.000}$	.778$_{\pm.002}$	.823$_{\pm.000}$	.949$_{\pm.001}$	.868$_{\pm.002}$	.659$_{\pm.003}$	.628$_{\pm.001}$	.644$_{\pm.002}$	.414$_{\pm.037}$	.773$_{\pm.005}$
UNI-2/392	350k	.827$_{\pm.006}$	**.928**$_{\pm.002}$	**.667**$_{\pm.003}$	.810$_{\pm.001}$	**.967**$_{\pm.000}$	.787$_{\pm.003}$	**.850**$_{\pm.000}$	.929$_{\pm.001}$	**.882**$_{\pm.004}$	.664$_{\pm.003}$	.629$_{\pm.001}$	.659$_{\pm.001}$	.424$_{\pm.041}$	**.771**$_{\pm.005}$
UNI-2/392 [CLS]	350k	.821$_{\pm.006}$	.917$_{\pm.003}$	.663$_{\pm.003}$	.829$_{\pm.001}$	.965$_{\pm.000}$	.791$_{\pm.002}$	.849$_{\pm.001}$	.927$_{\pm.001}$	.858$_{\pm.003}$	.653$_{\pm.004}$	.629$_{\pm.001}$	.659$_{\pm.001}$	.407$_{\pm.043}$	.767$_{\pm.005}$
Midnight-92k	92k	**.882**$_{\pm.005}$	.889$_{\pm.003}$	.615$_{\pm.003}$	.793$_{\pm.001}$	**.967**$_{\pm.000}$	**.823**$_{\pm.001}$	.831$_{\pm.000}$	.948$_{\pm.000}$	**.872**$_{\pm.004}$	.643$_{\pm.003}$	.629$_{\pm.000}$	.656$_{\pm.003}$	**.425**$_{\pm.032}$	.767$_{\pm.004}$
Midnight-92k [CLS]	92k	.876$_{\pm.006}$	.896$_{\pm.002}$	.616$_{\pm.004}$	.789$_{\pm.001}$	.966$_{\pm.000}$	.820$_{\pm.002}$	.811$_{\pm.000}$	.950$_{\pm.000}$	.861$_{\pm.005}$	.625$_{\pm.003}$	.629$_{\pm.000}$	.656$_{\pm.002}$	.392$_{\pm.039}$	.761$_{\pm.005}$
Virchow2	3.1M	.835$_{\pm.006}$	.890$_{\pm.004}$	.633$_{\pm.003}$	.818$_{\pm.001}$	.966$_{\pm.000}$	.791$_{\pm.004}$	**.865**$_{\pm.001}$	.938$_{\pm.001}$	.860$_{\pm.001}$	.646$_{\pm.003}$	.640$_{\pm.001}$	.674$_{\pm.002}$	.403$_{\pm.046}$	.766$_{\pm.006}$
Virchow2 [CLS]	3.1M	.851$_{\pm.007}$	.884$_{\pm.004}$	.624$_{\pm.003}$	.823$_{\pm.004}$	.966$_{\pm.000}$	.778$_{\pm.006}$	.861$_{\pm.000}$	.936$_{\pm.001}$	.865$_{\pm.001}$	.656$_{\pm.004}$	.639$_{\pm.001}$	.676$_{\pm.002}$	.398$_{\pm.040}$	.766$_{\pm.006}$
Midnight-12k	12k	.803$_{\pm.007}$	**.907**$_{\pm.001}$	.639$_{\pm.002}$	.840$_{\pm.000}$	**.967**$_{\pm.000}$	.790$_{\pm.001}$	.815$_{\pm.000}$	.931$_{\pm.000}$	.869$_{\pm.003}$	.656$_{\pm.003}$	.625$_{\pm.000}$	.664$_{\pm.001}$	.412$_{\pm.036}$	.763$_{\pm.004}$
Midnight-12k [CLS]	12k	.791$_{\pm.007}$	.904$_{\pm.001}$	.644$_{\pm.001}$	.841$_{\pm.000}$	.966$_{\pm.000}$	.801$_{\pm.001}$	.807$_{\pm.000}$	.930$_{\pm.000}$	.850$_{\pm.004}$	.663$_{\pm.002}$	.626$_{\pm.001}$	.663$_{\pm.001}$	.395$_{\pm.040}$	.760$_{\pm.005}$
Kaiko-B8	29k	.799$_{\pm.007}$	.876$_{\pm.004}$	.641$_{\pm.004}$	**.842**$_{\pm.003}$	.960$_{\pm.000}$	.761$_{\pm.006}$	.830$_{\pm.000}$	.920$_{\pm.000}$	.836$_{\pm.001}$	.650$_{\pm.003}$	**.644**$_{\pm.000}$	.686$_{\pm.001}$	.391$_{\pm.048}$	.757$_{\pm.006}$
Kaiko-B8 [CLS]	29k	.786$_{\pm.008}$	.872$_{\pm.006}$	.617$_{\pm.005}$	.825$_{\pm.004}$	.957$_{\pm.000}$	.748$_{\pm.004}$	.828$_{\pm.001}$	.917$_{\pm.001}$	.831$_{\pm.006}$	.642$_{\pm.003}$	.643$_{\pm.001}$	.686$_{\pm.001}$	.373$_{\pm.052}$	.748$_{\pm.007}$
tcga-100M	12k	.789$_{\pm.007}$	.873$_{\pm.002}$	.619$_{\pm.002}$	.814$_{\pm.001}$	**.968**$_{\pm.000}$	**.798**$_{\pm.001}$	.808$_{\pm.000}$	.928$_{\pm.000}$	.870$_{\pm.004}$	**.675**$_{\pm.003}$	.622$_{\pm.000}$	.656$_{\pm.002}$	.415$_{\pm.038}$	.757$_{\pm.005}$
tcga-100M [CLS]	12k	.774$_{\pm.007}$	.864$_{\pm.001}$	.615$_{\pm.001}$	.779$_{\pm.000}$	.967$_{\pm.000}$	.799$_{\pm.000}$	.792$_{\pm.000}$	.927$_{\pm.000}$	.852$_{\pm.007}$	.667$_{\pm.005}$	.622$_{\pm.001}$	.656$_{\pm.001}$	.396$_{\pm.041}$	.747$_{\pm.005}$
H-Optimus-0	500k	.831$_{\pm.007}$	.752$_{\pm.004}$	.620$_{\pm.006}$	.813$_{\pm.003}$	.962$_{\pm.000}$	.769$_{\pm.004}$	**.850**$_{\pm.001}$	.943$_{\pm.001}$	.847$_{\pm.003}$	**.672**$_{\pm.003}$	**.644**$_{\pm.002}$	**.687**$_{\pm.001}$	**.425**$_{\pm.037}$	.755$_{\pm.006}$
H-Optimus-0 [CLS]	500k	.824$_{\pm.007}$	.757$_{\pm.003}$	.615$_{\pm.004}$	.808$_{\pm.004}$	.956$_{\pm.002}$	.771$_{\pm.003}$	.842$_{\pm.002}$	.942$_{\pm.001}$	.838$_{\pm.004}$	.670$_{\pm.004}$	.644$_{\pm.001}$	.685$_{\pm.002}$	.415$_{\pm.038}$	.751$_{\pm.006}$
Prov_GigaPath	171k	.853$_{\pm.006}$	.794$_{\pm.002}$	.626$_{\pm.002}$	**.846**$_{\pm.001}$	.959$_{\pm.000}$	.727$_{\pm.004}$	.831$_{\pm.000}$	.944$_{\pm.000}$	.812$_{\pm.004}$	.657$_{\pm.003}$	.628$_{\pm.001}$	**.688**$_{\pm.001}$	.405$_{\pm.045}$	.752$_{\pm.005}$
Prov_GigaPath [CLS]	171k	.852$_{\pm.006}$	.766$_{\pm.004}$	.616$_{\pm.002}$	.821$_{\pm.002}$	.951$_{\pm.000}$	.720$_{\pm.005}$	.831$_{\pm.001}$	.942$_{\pm.001}$	.791$_{\pm.006}$	.660$_{\pm.003}$	.626$_{\pm.002}$	.687$_{\pm.002}$	.393$_{\pm.047}$	.743$_{\pm.006}$
Hibou-L	1.1M	.825$_{\pm.006}$	.792$_{\pm.004}$	.643$_{\pm.005}$	.767$_{\pm.002}$	.954$_{\pm.001}$	.766$_{\pm.001}$	**.850**$_{\pm.001}$	**.949**$_{\pm.001}$	.852$_{\pm.003}$	.654$_{\pm.002}$	**.646**$_{\pm.001}$	.668$_{\pm.001}$	.397$_{\pm.052}$	.751$_{\pm.006}$
Hibou-L [CLS]	1.1M	.804$_{\pm.007}$	.811$_{\pm.006}$	.637$_{\pm.007}$	.740$_{\pm.001}$	.933$_{\pm.002}$	.763$_{\pm.001}$	.839$_{\pm.000}$	.952$_{\pm.001}$	.823$_{\pm.003}$	.634$_{\pm.005}$	.645$_{\pm.001}$	.668$_{\pm.001}$	.388$_{\pm.048}$	.741$_{\pm.007}$
UNI	100k	.833$_{\pm.006}$	.797$_{\pm.006}$	.613$_{\pm.003}$	.808$_{\pm.004}$	.954$_{\pm.001}$	.759$_{\pm.007}$	.841$_{\pm.001}$	.937$_{\pm.002}$	.854$_{\pm.003}$	.662$_{\pm.004}$	.627$_{\pm.001}$	.662$_{\pm.003}$	.391$_{\pm.049}$	.749$_{\pm.007}$
UNI [CLS]	100k	.815$_{\pm.007}$	.791$_{\pm.009}$	.593$_{\pm.001}$	.789$_{\pm.005}$	.948$_{\pm.002}$	.757$_{\pm.005}$	.840$_{\pm.001}$	.938$_{\pm.001}$	.822$_{\pm.006}$	.655$_{\pm.004}$	.627$_{\pm.000}$	.659$_{\pm.004}$	.386$_{\pm.051}$	.740$_{\pm.007}$
UNI/512	100k	.755$_{\pm.008}$	.891$_{\pm.004}$	.624$_{\pm.004}$	.751$_{\pm.004}$	.951$_{\pm.001}$	.757$_{\pm.007}$	.831$_{\pm.000}$	.890$_{\pm.001}$	.811$_{\pm.005}$	.647$_{\pm.002}$	.620$_{\pm.001}$	.671$_{\pm.003}$	.380$_{\pm.045}$	.737$_{\pm.007}$
UNI/512 [CLS]	100k	.737$_{\pm.008}$	.877$_{\pm.008}$	.612$_{\pm.001}$	.732$_{\pm.004}$	.950$_{\pm.000}$	.754$_{\pm.010}$	.814$_{\pm.002}$	.883$_{\pm.001}$	.814$_{\pm.006}$	.654$_{\pm.002}$	.621$_{\pm.001}$	.658$_{\pm.003}$	.364$_{\pm.046}$	.728$_{\pm.007}$
Phikon	12k	.826$_{\pm.007}$	.744$_{\pm.003}$	.579$_{\pm.002}$	.715$_{\pm.006}$	.946$_{\pm.001}$	.743$_{\pm.005}$	.824$_{\pm.000}$	.919$_{\pm.002}$	.822$_{\pm.006}$	.648$_{\pm.003}$	.624$_{\pm.001}$	.644$_{\pm.002}$	.377$_{\pm.048}$	.724$_{\pm.007}$
Phikon [CLS]	12k	.820$_{\pm.007}$	.735$_{\pm.004}$	.568$_{\pm.015}$	.713$_{\pm.004}$	.942$_{\pm.002}$	.729$_{\pm.003}$	.804$_{\pm.001}$	.923$_{\pm.000}$	.809$_{\pm.008}$	.644$_{\pm.004}$	.623$_{\pm.001}$	.644$_{\pm.003}$	.367$_{\pm.050}$	.717$_{\pm.008}$
Phikon-v2	60k	.756$_{\pm.007}$	.737$_{\pm.004}$	.607$_{\pm.003}$	.725$_{\pm.002}$	.953$_{\pm.000}$	.753$_{\pm.001}$	.796$_{\pm.000}$	.900$_{\pm.000}$	.807$_{\pm.003}$	.634$_{\pm.003}$	.626$_{\pm.000}$	.645$_{\pm.003}$	.391$_{\pm.045}$	.718$_{\pm.006}$
Phikon-v2 [CLS]	60k	.741$_{\pm.008}$	.734$_{\pm.003}$	.600$_{\pm.003}$	.716$_{\pm.003}$	.939$_{\pm.001}$	.755$_{\pm.002}$	.784$_{\pm.000}$	.893$_{\pm.000}$	.803$_{\pm.003}$	.631$_{\pm.002}$	.626$_{\pm.001}$	.645$_{\pm.003}$	.375$_{\pm.041}$	.711$_{\pm.005}$
Lunit	36k	.763$_{\pm.007}$	.785$_{\pm.006}$	.627$_{\pm.002}$	.759$_{\pm.005}$	.943$_{\pm.001}$	.758$_{\pm.004}$	.785$_{\pm.000}$	.905$_{\pm.001}$	.759$_{\pm.011}$	.604$_{\pm.003}$	.600$_{\pm.000}$	.630$_{\pm.001}$	.362$_{\pm.063}$	.714$_{\pm.008}$
Lunit [CLS]	36k	.753$_{\pm.008}$	.782$_{\pm.008}$	.614$_{\pm.004}$	.750$_{\pm.007}$	.938$_{\pm.001}$	.747$_{\pm.003}$	.779$_{\pm.001}$	.901$_{\pm.001}$	.730$_{\pm.006}$	.610$_{\pm.004}$	.599$_{\pm.001}$	.629$_{\pm.001}$	.353$_{\pm.054}$	.707$_{\pm.008}$
vitg14 (nat. img.)	0	.721$_{\pm.006}$	.724$_{\pm.002}$	.578$_{\pm.003}$	.783$_{\pm.001}$	.943$_{\pm.000}$	.740$_{\pm.002}$	**.855**$_{\pm.001}$	.881$_{\pm.001}$	.500$_{\pm.010}$	.509$_{\pm.006}$	.565$_{\pm.001}$	.614$_{\pm.001}$	.351$_{\pm.042}$	.674$_{\pm.006}$
vitg14 (nat. img.) [CLS]	0	.719$_{\pm.005}$	.725$_{\pm.003}$	.583$_{\pm.002}$	.832$_{\pm.001}$	.935$_{\pm.000}$	.744$_{\pm.002}$	.862$_{\pm.001}$	.874$_{\pm.001}$	.507$_{\pm.004}$	.382$_{\pm.016}$	.564$_{\pm.001}$	.614$_{\pm.001}$	.342$_{\pm.043}$	.668$_{\pm.006}$
vitg14 (initial)	0	.652$_{\pm.006}$	.474$_{\pm.005}$	.413$_{\pm.007}$	.425$_{\pm.003}$	.754$_{\pm.003}$	.459$_{\pm.002}$	.578$_{\pm.004}$	.763$_{\pm.001}$	.526$_{\pm.008}$	.304$_{\pm.003}$	.462$_{\pm.001}$	.432$_{\pm.005}$	.166$_{\pm.087}$	.493$_{\pm.010}$
vitg14 (initial) [CLS]	0	.649$_{\pm.006}$	.473$_{\pm.002}$	.411$_{\pm.004}$	.427$_{\pm.008}$	.748$_{\pm.003}$	.464$_{\pm.003}$	.569$_{\pm.000}$	.755$_{\pm.002}$	.566$_{\pm.006}$	.308$_{\pm.002}$	.461$_{\pm.003}$	.428$_{\pm.004}$	.172$_{\pm.089}$	.495$_{\pm.010}$