Leveraging Model-Generated Annotations for Nuclei Segmentation in Computational Pathology

Christopher Hendra[1], Kelvin Chong[1], Charlene Ong[1], Michelle Nguyen Ngo[2], Chih-Liang Chin[3], Richard Baumgartner[2], Shubing Wang[2], Asad Abu Bakar Ali[1], and Desiree Abdurrachim[1]

[1]Quantitative Biosciences, MSD, Singapore, Singapore
[2]Biostatistics and Research Decision Sciences, Merck & Co., Inc., Rahway, NJ, USA
[3]Cardiometabolic Diseases, Merck & Co., Inc., South San Francisco, CA, USA

Abstract

Detecting and segmenting nuclei from hematoxylin and eosin (H&E) stained images is important for many downstream applications, ranging from disease diagnosis in clinical setting to biomarkers development in the preclinical setting. Many open-source models have been developed for cell segmentation on publicly available datasets, but they might not generalize well across different tissue and disease conditions and finetuning these models can be costly owing to the labour-intensive nature of annotating from H&E images. To address this, we propose a novel training framework that leverages annotations derived from multiple pre-existing segmentation models, treating them as imperfect "annotators". Our approach mitigates the risk of overfitting to the inherent biases of these source models by incorporating learnable embedding vectors that explicitly represent the distinct annotation "style" of each model. This allows our model to learn robust, generalizable features despite the limited availability of ground-truth annotations. We show that this approach results in a superior segmentation performance compared to naively training on the aggregated outputs of pre-trained models.

1 Introduction

Nuclei segmentation is a crucial step in the analysis of pathology images, providing important information such as cellular morphology and distribution that makes diagnosis and interpreting of disease biology possible. Alterations in nuclear morphology, membrane irregularities, and increased nuclear to cytoplasmic ratio, for example, are essential diagnostic features to distinguish benign from malignant cells Fischer [2020]. As such, a lot of efforts have been made in the field to develop robust segmentation algorithms and many publicly available datasets Kumar et al. [2019], Graham et al. [2019, 2021], Gamper et al. [2019, 2020], Graham et al. [2024], Lin et al. [2023] have enabled the continuous improvement of these algorithms over the years. Recently, owing to the growing availability of compute and datasets, Deep Learning (DL) based approaches have emerged as the dominant paradigm for nuclei segmentation. Models such as StarDist Schmidt et al. [2018], Hover-NetGraham et al. [2019], CellPose Stringer et al. [2021], and Instanseg Goldsborough et al. [2024] rely on some variant of the U-Net architecture Ronneberger et al. [2015] with multiple prediction heads, one for detecting pixels containing the nuclei and other heads for predicting distance related measures for each nucleus to separate them into different instances. More recent works such as CellVit Hörst et al. [2024] and CellVit++ Hörst et al. [2025] leverage large pretrained Vision Transformers (ViT)Dosovitskiy et al. [2020], Kirillov et al. [2023], Chen et al. [2022a], Vorontsov et al. [2023], Chen et al. [2024], Zimmermann et al. [2024] to achieve state-of-the-arts (SOTA) results on several different cell segmentation benchmarks.

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: The 3rd Workshop on Imageomics: Discovering Biological Knowledge from Images Using AI.

Despite these advancements, the generalization capability of these models remains a challenge. The identification of nuclei can be challenging due to the vast number of different cell types with diverse cellular morphologies across different tissue types and diseases, and the publicly available annotated datasets that these models are trained on, while valuable, are often specific in scope. Creating a comprehensive, manually annotated datasets that can capture the vast heterogeneity of cellular appearances across all relevant domains, however, is prohibitively expensive and labour-intensive. To address these limitations, we introduce a novel training framework designed to effectively learn from diverse, model-derived annotations while actively mitigating the propagation of source model biases. Here we frame the problem of learning from model-derived annotations similarly as learning from multiple annotators, a topic that has been explored extensively in medical imaging. Much prior work focuses on inferring a consensus label from annotators with different level of reliability, followed by model training on this fused segmentation. The STAPLE algorithm Warfield et al. [2004], is a popular method for estimating a true segmentation label by weighting each annotator's contribution per pixel using EM algorithm Dempster et al. [1977]. Subsequent works aimed to improve STAPLE, such as by modeling accuracy per class to minimize the contribution of background pixels Asman and Landman [2011] or by incorporating annotators' spatial variability Asman and Landman [2012]. In instance segmentation, Le Le et al. [2023] combined annotations using Weighted Box Fusion (WBF) Solovyev et al. [2021] and more recently, Zhang Zhang et al. [2023] proposed an end-to-end approach where consensus labels from different experts are learned jointly with the annotator reliability.

In contrast with existing approaches, our goal here is to decouple generalizable nuclear features from model-specific artefacts. Our approach incorporates a learnable embedding vector to capture the distinct "annotation style" of each pre-trained model. We show that this approach outperforms naively training on source model outputs especially in low data regimes.

2 Methods

2.1 Network Structure

Following the success of CellVit in integrating large foundation models with the HoverNet approach, we leverage Optimus H-0 Saillard et al. [2024] as the backbone of our network with the ViT adapter Chen et al. [2022b] module to inject spatial information from the input image into the ViT backbone to produce multi-scale feature maps. We adopt a similar strategy to HoVer-Net by having two decoder branches, each based on an U-Net shaped encoder-decoder architecture. The first branch (NP branch) predicts the binary segmentation map of all nuclei and the second branch (HV branch) predicts the horizontal and vertical distance maps from the center of each nucleus. Our training strategy follows CellVit closely where we have an additional tissue classification head (TC) based on the class token output from our ViT encoder and the the model is trained to minimize the weighted sum of the loss terms related to the output of each branch. More details can be found in the appendix of this paper.

During inference, we first obtain the binary segmentation map by thresholding the output of the NP branch. Afterwards, we separate the nuclei instances using HoVer-Net's postprocessing pipeline where we apply Sobel operator to the distance maps followed by marker-controlled watersheld algorithm to generate the individual nuclear boundary.

2.2 Learning from model-generated annotations

Most of the work involved in learning from multiple annotators aims to generate a consensus label by minimizing noise from multiple annotators. However, our approach focuses on leveraging model-generated annotations to enrich training data and improve segmentation performance, particularly when gold-standard annotations are scarce. Pre-trained models, often trained with different DNN architectures and training objectives, invariably learn distinct "styles" of feature projection and produce systematically different annotations. As shown in Figure 1, our architecture comprises three main components 1) a ViT encoder, 2) a ViT adapter module, 3) and a multi-branch decoder. Here, we modify our decoder to incorporate style-aware convolutional block to capture the different annotation styles in our datasets.

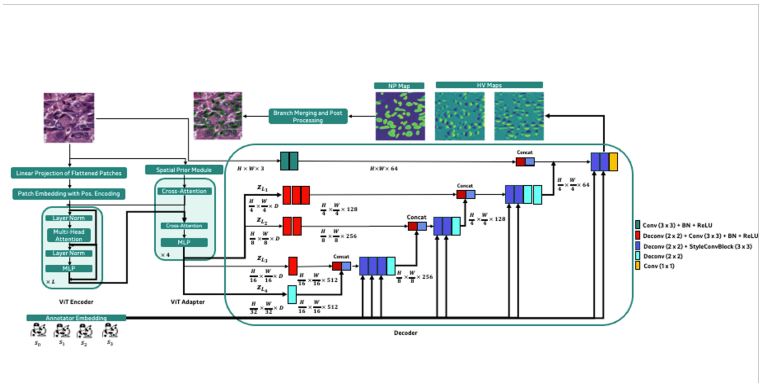

Figure 1: The architecture of our multi-annotation approach. An input H&E image is processed by a ViT backbone to extract multi-scale features (z_{L_k}). In the U-Net-like decoder, feature maps are modulated by a StyleConvBlock conditioned on a style embedding (s_j) that is unique for each annotator. During training, annotator embedding s_j is determined by a randomly sampled annotation source (human or model). During inference, the embedding for the human ground truth (s_0) is used exclusively. The decoder produces Nuclear Pixel (NP) and Horizontal/Vertical (HV) maps, which are post-processed to yield the final instance segmentation.

Formally, given an input image x, our encoder outputs a multi-scale feature map $z = \{z_{L_1}, z_{L_2}, z_{L_3}, z_{L_4}\}$ where $z_{L_k} \in \mathbb{R}^{\frac{H}{2^{k+1}} \times \frac{W}{2^{k+1}} \times D_k}$ for $k = 1, 2, 3, 4$ (level L_4 being the coarsest). The decoder reconstructs the segmentation map through a series of deconvolution blocks. The output feature map of the k-th decoder block f_k is given by:

$$f_k(x) = \mathcal{G}_k(\text{Concat}(f_{k-1}(x), z_{L_k})) \tag{1}$$
$$\mathcal{G}_k(x) = \text{Upsample}(\text{ConvBlock}(\text{ConvBlock}(x))) \tag{2}$$
$$\text{ConvBlock}(x) = \text{ReLU}(\text{BatchNorm}(\text{Conv}(x))) \tag{3}$$

Here, $\mathcal{G}_k$ typically comprises two blocks of 3×3 convolutional layer followed by BatchNorm and ReLu layer with one deconvolutional layer to upsample the output feature map at the end.

We have access to pixel level ground truth $y^j \in \mathbb{R}^{H \times W}$ from a set of N_A annotators (source models) plus, for some images, a gold-standard human annotation y^0. Our goal is to enable the segmentation model to distinguish annotator-specific characteristics from robust features that are genuinely indicative of the presence of nuclei in the input image. We propose to modulate the features within each processing block $\mathcal{G}_k$ based on the annotator j whose annotation y^j is used for supervision. We modify $\mathcal{G}_k$ to $\tilde{\mathcal{G}}_k$ by replacing ConvBlock with the style-aware StyleConvBlock as follows:

$$\tilde{\mathcal{G}}_k(x, s_j) = \text{Upsample}(\text{StyleConvBlock}(\text{StyleConvBlock}(x, s_j))) \tag{4}$$
$$\text{StyleConvBlock}(x, s_j) = \text{ReLU}(\text{BatchNorm}(\alpha_\theta(s_j) \cdot \text{Conv}(x) + \gamma_\phi(s_j))) \tag{5}$$

Here s_j is a learnable embedding vector for annotator j. The functions $\alpha_\theta(\cdot)$ and $\gamma_\phi(\cdot)$ are small neural networks (e.g., linear layers) with parameters θ and ϕ respectively, which transform s_j into channel-wise scale and shift parameters. This transformation applies annotator-conditional affine transformation to the convolutional features before batch normalization and ReLU activation.

3 Results

Our primary objective is to evaluate the contribution of the StyleConvBlock in learning robust feature representations from noisy, model-generated annotations. To this end, we designed experiments to

test our framework on the PanNuke, a dataset that contains 7,901 images, with over 189,744 labeled nuclei, following the split in Hörst et al. [2024]. We then simulate two challenging scenarios on this dataset: 1) a zero ground-truth scenario, where only model-generated labels are available for training, and 2) a limited-annotation scenario, where only a small fraction of ground-truth data is available.

Table 1: Performance analysis of model finetuning on the outputs of three source models, Stardist Schmidt et al. [2018], Cellpose Stringer et al. [2021], and Instanseg Goldsborough et al. [2024] model on the PanNuke dataset against finetuning pretrained ViT on the source model outputs and our approach (Multi-annotators) conditioned on the embedding style of each pretrained model

Source Model	Model	bPQ	Precision	Recall	F1
Stardist	Stardist	0.570	0.708	**0.798**	0.750
	Finetuned ViT	0.592	0.732	0.795	**0.767**
	Multi-annotations Finetuned ViT	**0.598**	**0.738**	0.794	0.765
Instanseg	Instanseg	0.581	0.810	0.712	0.758
	Finetuned ViT	0.593	**0.817**	0.715	0.762
	Multi-annotations Finetuned ViT	**0.602**	0.816	**0.731**	**0.771**
Cellpose	Cellpose	0.543	0.740	0.706	0.723
	Finetuned ViT	0.555	0.774	0.695	0.732
	Multi-annotations Finetuned ViT	**0.569**	**0.779**	**0.708**	**0.742**
Full Groundtruth		0.661	0.832	0.784	0.807

3.1 Performance in a Zero-Annotation Setting

First, we assessed whether our framework could improve performance without access to any human-annotated ground-truth labels. We used annotations generated by three source models (Stardist, Instanseg, Cellpose) for training and evaluate the performance of the model on held-out ground-truth annotations. As shown in Table 1, simply finetuning a pre-trained ViT on the outputs of a single source model (Finetuned ViT) consistently improves the performance over the source model. The Finetuned ViT achieves a bPQ of 0.592, 0.593 and 0.555, surpassing the source Stardist, Instanseg, and Cellpose models' bPQ of 0.570, 0.581, and 0.543. Crucially, when training a single network on annotations from all three models using our proposed multi-annotation framework, performance is enhanced further. By conditioning the network on the specific "style" embedding of a source model, our Multi-annotations Finetuned ViT outperforms both the original source model and the naively finetuned ViT across the board. For instance, when conditioned on the Instanseg style, our model achieves a bPQ of 0.602, higher than both the source Instanseg (0.581) and naively finetuned model (0.593). These results suggest that the StyleConvBlock is able to disentagle model-specific artifacts from generalizable nuclear features, allowing for a more robust learning even when ground-truth annotation is absent

3.2 Performance in a Limited-Annotation Setting

To simulate a more realistic scenario where annotation budgets are limited, we evaluated performance when only a small fraction (5% to 25%) of the ground-truth data is available, supplementing the rest with model-generated annotations. We repeat the experiment 5 times for each fold to account for the variability in model performance due to sampling and we compared three training strategies:

1. GT only: Training solely on the available subset of the ground-truth annotations
2. GT + Instanseg/Stardist/Cellpose/All : Naively mixing the ground-truth subset with annotations from the source models. In GT + All, we randomly sample the source annotation for each training pass, just like the multi-annotations approach but without incorporating annotator embedding vector to the model.
3. Multi-annotations: Our proposed framework, using the ground-truth subset and annotations from all three source models. Inference is done conditioned on the ground-truth style embedding.

The results, summarized in Table 2 demonstrate the superiority of our multi-annotation framework, particularly in low-data regimes. At just 5% of ground-truth availability, our model achieves a bPQ of

Table 2: Model performance Metrics at Different Training Fractions. For each fraction we compare training on a mixture of a source model and ground-truth annotations against training only on images with annotations and training using the multi-annotations approach

Training Label %	Model	bPQ	Precision	Recall	F1
5	GT + Instanseg	0.596	**0.780**	0.758	0.769
	GT + Stardist	0.583	0.709	0.808	0.755
	GT + Cellpose	0.561	0.738	0.735	0.736
	GT + All	0.604	0.769	0.784	0.777
	GT only	0.571	0.715	0.774	0.743
	Multi-annotators	**0.620**	0.776	**0.790**	**0.783**
10	GT + Instanseg	0.594	0.776	0.759	0.767
	GT + Stardist	0.584	0.710	0.809	0.756
	GT + Cellpose	0.564	0.743	0.635	0.739
	GT + All	0.604	0.771	0.784	0.777
	GT only	0.595	0.745	**0.784**	0.764
	Multi-annotators	**0.627**	**0.796**	0.783	**0.789**
15	GT + Instanseg	0.596	0.782	0.756	0.769
	GT + Stardist	0.586	0.712	0.809	0.757
	GT + Celpose	0.561	0.736	0.736	0.736
	GT only	0.606	0.763	0.792	0.777
	Multi-annotators	**0.631**	**0.796**	**0.791**	**0.793**
20	GT + Instanseg	0.594	0.786	0.750	0.767
	GT + Stardist	0.584	0.706	0.810	0.754
	GT + Cellpose	0.558	0.740	0.732	0.736
	GT + All	0.606	0.765	0.791	0.778
	GT only	0.615	0.776	0.782	0.779
	Multi-annotators	**0.635**	**0.804**	**0.788**	**0.796**
25	GT + Instanseg	0.593	0.783	0.753	0.768
	GT + Stardist	0.584	0.706	**0.810**	0.754
	GT + Cellpose	0.562	0.742	0.734	0.738
	GT + All	0.609	0.769	0.790	0.779
	GT only	0.622	0.779	0.789	0.784
	Multi-annotators	**0.633**	**0.805**	0.785	**0.795**

0.620 and and F1-score of 0.783. This is a dramatic improvement over training on the ground-truth subset alone (bPQ 0.571, F1 0.743) and naively mixing in model-generated labels. Furthermore, the performance of the naively finetuned ViT model fails to improve as more ground-truth data is added, indicating that the network is overfitting to the noisy, model-generated labels rather than learning from the sparse, high quality ground-truth. In contrast, our framework effectively leverages both sources of information, consistently outperforming the other approaches across all tested fractions of training annotations. This highlights the robustness of our proposed framework and its ability to mitigate the negative impact of noisy labels during training.

4 Discussion

Nuclear segmentation in H&E stained images remains a foundational yet challenging task in computational pathology. The vast diversity in cellular morphology across different tissues and diseases, coupled with the high cost of generating expert annotations, limits the generalization capabilities of existing models. To address this, we proposed a novel training framework that incorporates a style-aware StyleConvBlock into a ViT-based segmentation network. This approach effectively leverages annotations from multiple pre-existing models, treating them as imperfect annotators with distinct "styles". Our approach outperforms naively finetuning on pre-existing models' outputs and shows a strong baseline performance in low data regimes. Our framework shows a promise in providing a cost-effective strategy to develop a more-customized computational pathology tools, accelerating both research and clinical applications. Future work will include improvement and validation of the framework across different datasets as well as its effectiveness in improving model performance when more annotations are available.

References

Andrew J Asman and Bennett A Landman. Robust statistical label fusion through consensus level, labeler accuracy, and truth estimation (collate). *IEEE transactions on medical imaging*, 30(10): 1779–1794, 2011.

Andrew J Asman and Bennett A Landman. Formulating spatially varying performance in the statistical fusion framework. *IEEE transactions on medical imaging*, 31(6):1326–1336, 2012.

Richard J Chen, Chengkuan Chen, Yicong Li, Tiffany Y Chen, Andrew D Trister, Rahul G Krishnan, and Faisal Mahmood. Scaling vision transformers to gigapixel images via hierarchical self-supervised learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16144–16155, 2022a.

Richard J Chen, Tong Ding, Ming Y Lu, Drew FK Williamson, Guillaume Jaume, Andrew H Song, Bowen Chen, Andrew Zhang, Daniel Shao, Muhammad Shaban, et al. Towards a general-purpose foundation model for computational pathology. *Nature Medicine*, 30(3):850–862, 2024.

Zhe Chen, Yuchen Duan, Wenhai Wang, Junjun He, Tong Lu, Jifeng Dai, and Yu Qiao. Vision transformer adapter for dense predictions. *arXiv preprint arXiv:2205.08534*, 2022b.

Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the royal statistical society: series B (methodological)*, 39(1): 1–22, 1977.

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.

Edgar G Fischer. Nuclear morphology and the biology of cancer cells. *Acta cytologica*, 64(6): 511–519, 2020.

Jevgenij Gamper, Navid Alemi Koohbanani, Ksenija Benes, Ali Khuram, and Nasir Rajpoot. Pannuke: an open pan-cancer histology dataset for nuclei instance segmentation and classification. In *European Congress on Digital Pathology*, pages 11–19. Springer, 2019.

Jevgenij Gamper, Navid Alemi Koohbanani, Ksenija Benes, Simon Graham, Mostafa Jahanifar, Syed Ali Khurram, Ayesha Azam, Katherine Hewitt, and Nasir Rajpoot. Pannuke dataset extension, insights and baselines. *arXiv preprint arXiv:2003.10778*, 2020.

Thibaut Goldsborough, Ben Philps, Alan O'Callaghan, Fiona Inglis, Leo Leplat, Andrew Filby, Hakan Bilen, and Peter Bankhead. Instanseg: an embedding-based instance segmentation algorithm optimized for accurate, efficient and portable cell segmentation. *arXiv preprint arXiv:2408.15954*, 2024.

Simon Graham, Quoc Dang Vu, Shan E Ahmed Raza, Ayesha Azam, Yee Wah Tsang, Jin Tae Kwak, and Nasir Rajpoot. Hover-net: Simultaneous segmentation and classification of nuclei in multi-tissue histology images. *Medical image analysis*, 58:101563, 2019.

Simon Graham, Mostafa Jahanifar, Ayesha Azam, Mohammed Nimir, Yee-Wah Tsang, Katherine Dodd, Emily Hero, Harvir Sahota, Atisha Tank, Ksenija Benes, et al. Lizard: a large-scale dataset for colonic nuclear instance segmentation and classification. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 684–693, 2021.

Simon Graham, Quoc Dang Vu, Mostafa Jahanifar, Martin Weigert, Uwe Schmidt, Wenhua Zhang, Jun Zhang, Sen Yang, Jinxi Xiang, Xiyue Wang, et al. Conic challenge: Pushing the frontiers of nuclear detection, segmentation, classification and counting. *Medical image analysis*, 92:103047, 2024.

Fabian Hörst, Moritz Rempe, Lukas Heine, Constantin Seibold, Julius Keyl, Giulia Baldini, Selma Ugurel, Jens Siveke, Barbara Grünwald, Jan Egger, et al. Cellvit: Vision transformers for precise cell segmentation and classification. *Medical Image Analysis*, 94:103143, 2024.

Fabian Hörst, Moritz Rempe, Helmut Becker, Lukas Heine, Julius Keyl, and Jens Kleesiek. Cellvit++: Energy-efficient and adaptive cell segmentation and classification using foundation models. *arXiv preprint arXiv:2501.05269*, 2025.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3, 2022.

Alexander Kirillov, Kaiming He, Ross Girshick, Carsten Rother, and Piotr Dollár. Panoptic segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9404–9413, 2019.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4015–4026, 2023.

Neeraj Kumar, Ruchika Verma, Deepak Anand, Yanning Zhou, Omer Fahri Onder, Efstratios Tsougenis, Hao Chen, Pheng-Ann Heng, Jiahui Li, Zhiqiang Hu, et al. A multi-organ nucleus segmentation challenge. *IEEE transactions on medical imaging*, 39(5):1380–1391, 2019.

Khiem H Le, Tuan V Tran, Hieu H Pham, Hieu T Nguyen, Tung T Le, and Ha Q Nguyen. Learning from multiple expert annotators for enhancing anomaly detection in medical image analysis. *IEEE Access*, 11:14105–14114, 2023.

Yi Lin, Zhiyong Qu, Hao Chen, Zhongke Gao, Yuexiang Li, Lili Xia, Kai Ma, Yefeng Zheng, and Kwang-Ting Cheng. Nuclei segmentation with point annotations from pathology images via self-supervised learning and co-training. *Medical Image Analysis*, 89:102933, 2023.

Sourab Mangrulkar, Sylvain Gugger, Lysandre Debut, Younes Belkada, Sayak Paul, and Benjamin Bossan. Peft: State-of-the-art parameter-efficient fine-tuning methods. `https://github.com/huggingface/peft`, 2022.

A Paszke. Pytorch: An imperative style, high-performance deep learning library. *arXiv preprint arXiv:1912.01703*, 2019.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, pages 234–241. Springer, 2015.

Charlie Saillard, Rodolphe Jenatton, Felipe Llinares-López, Zelda Mariet, David Cahané, Eric Durand, and Jean-Philippe Vert. H-optimus-0, 2024. URL `https://github.com/bioptimus/releases/tree/main/models/h-optimus/v0`.

Uwe Schmidt, Martin Weigert, Coleman Broaddus, and Gene Myers. Cell detection with star-convex polygons. In *Medical image computing and computer assisted intervention–MICCAI 2018: 21st international conference, Granada, Spain, September 16-20, 2018, proceedings, part II 11*, pages 265–273. Springer, 2018.

Roman Solovyev, Weimin Wang, and Tatiana Gabruseva. Weighted boxes fusion: Ensembling boxes from different object detection models. *Image and Vision Computing*, 107:104117, 2021.

Carsen Stringer, Tim Wang, Michalis Michaelos, and Marius Pachitariu. Cellpose: a generalist algorithm for cellular segmentation. *Nature methods*, 18(1):100–106, 2021.

Eugene Vorontsov, Alican Bozkurt, Adam Casson, George Shaikovski, Michal Zelechowski, Siqi Liu, Kristen Severson, Eric Zimmermann, James Hall, Neil Tenenholtz, et al. Virchow: A million-slide digital pathology foundation model. *arXiv preprint arXiv:2309.07778*, 2023.

Simon K Warfield, Kelly H Zou, and William M Wells. Simultaneous truth and performance level estimation (staple): an algorithm for the validation of image segmentation. *IEEE transactions on medical imaging*, 23(7):903–921, 2004.

Le Zhang, Ryutaro Tanno, Moucheng Xu, Yawen Huang, Kevin Bronik, Chen Jin, Joseph Jacob, Yefeng Zheng, Ling Shao, Olga Ciccarelli, et al. Learning from multiple annotators for medical image segmentation. *Pattern Recognition*, 138:109400, 2023.

Eric Zimmermann, Eugene Vorontsov, Julian Viret, Adam Casson, Michal Zelechowski, George Shaikovski, Neil Tenenholtz, James Hall, David Klimstra, Razik Yousfi, et al. Virchow2: Scaling self-supervised mixed magnification models in pathology. *arXiv preprint arXiv:2408.00738*, 2024.

5 Appendix

5.1 Training details

We optimize our network to minimize the following loss function:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{NP} + \mathcal{L}_{HV} + \mathcal{L}_{TC} \tag{6}$$

Here $\mathcal{L}_{NP}$ denotes the binary segmentation loss, $\mathcal{L}_{HV}$ denotes the horizontal and vertical distance loss and $\mathcal{L}_{TC}$ denotes the tissue type classification loss. The losses are composed of the following weighted loss functions:

$$\mathcal{L}_{NP} = \lambda_{NP_{\text{FT}}}\mathcal{L}_{\text{FT}} + \lambda_{NP_{\text{DICE}}}\mathcal{L}_{\text{DICE}} \tag{7}$$
$$\mathcal{L}_{HV} = \lambda_{HV_{\text{MSE}}}\mathcal{L}_{\text{MSE}} + \lambda_{HV_{\text{MSGE}}}\mathcal{L}_{\text{MSGE}} \tag{8}$$
$$\mathcal{L}_{TC} = \lambda_{\text{TC}}\mathcal{L}_{\text{CE}} \tag{9}$$

where $\mathcal{L}_{\text{TE}}$ is the Focal-Tversky loss, $\mathcal{L}_{\text{DICE}}$ is the dice loss, $\mathcal{L}_{\text{MSE}}$ is the mean squared error of the horizontal and vertical distance maps, $\mathcal{L}_{\text{MSGE}}$ is the mean squared error of the gradients of the horizontal and vertical distance maps and $\mathcal{L}_{\text{CE}}$ is the cross entropy loss for the tissue classification.

We trained our model over 100 epochs using the Pytorch library Paszke [2019] with LoRA Hu et al. [2022] as implemented by HuggingFace PEFT library Mangrulkar et al. [2022]. Training is performed with a batch size of 16 on 8 x NVIDIA A100 (40 GB) GPUs with 0.0005 learning rate and learning rate decay of 0.9. We also followed the weights used in Hörst et al. [2024] for our loss function. Our training dataset contains images with annotations from up to 4 sources: one human-generated ground truth (denoted $j = 0$) and 3 model-generated annotations Schmidt et al. [2018], Stringer et al. [2021], Goldsborough et al. [2024](denoted $j = 1, 2, 3$). For each image during a training iteration, we randomly sample one annotation source y^j from the available sources for that image. The corresponding annotator embedding s_j is then used in the style modulation layers (Eq. 5). The loss weights $\lambda_{NP_{\text{CE}}}, \lambda_{NP_{\text{DICE}}}, \lambda_{HV_{\text{MSE}}}, \lambda_{HV_{\text{MSGE}}}$ and λ_{TC} as Hörst et al. [2024].

5.2 Metrics

To assess the quality of nuclear instance segmentation, we use the binary panoptic quality (bPQ) Kirillov et al. [2019] metrics. The bPQ is defined as

$$bPQ = \frac{|TP|}{|TP| + \frac{1}{2}|FP| + \frac{1}{2}|FN|} \times \frac{\sum_{(y,\hat{y})\in TP} IoU(y, \hat{y})}{|TP|} \tag{10}$$

with IoU denoting the intersection over unionKirillov et al. [2019], y denoting the ground truth segment and $\hat{y}$ denoting the model prediction, and the pair $(y, \hat{y})$ being a unique matching prediction between ground truth segment and model prediction with a minimum IoU of 0.5. The True Positives $|TP|$ is the total number of matched pairs of segments, the False Positives $|FP|$ is the total number of predicted segments that are not matched with any ground truth segments, and False Negatives $|FN|$ is the total number of ground truth nuclei without matching predicted segments. Furthermore, we also compare the model predictions in terms of their precision Pr, recall R, and F_1 score as follows:

$$F_1 = \frac{2|TP|}{2|TP| + |FP| + |FN|} \tag{11}$$
$$Pr = \frac{|TP|}{|TP| + |FP|} \tag{12}$$
$$R = \frac{|TP|}{|TP| + |FN|} \tag{13}$$

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The main claims made in the abstract are about the improvement in the model performance by adding style layers to incorporate model-generated annotations over naive training using model-generated annotations and training with limited annotations. This is supported by our experimental results.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss the limitations of this work in the discussion section where we highlight that future work will have to expand the scope of the validation of this work across more datasets.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

   Justification: We do not have any theoretical results in this paper

   Guidelines:

   - The answer NA means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Yes we fully disclose the information needed to reproduce the main experimental results. We provide the hyper-parameters for this experiment and all model architectures and datasets are publicly available.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility.

In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The model and data are publicly available. The code will be made available upon request.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: They are specified in the paper.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: We do not have enough time to generate the error bars needed for this question for this submission. We observed that the differences in the metrics we measured in the experiment were sufficiently large and the experiments for the main results on table 2 and each experiment was repeated 15 times. We can provide more details if needed during the rebuttal period.

   Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Yes the paper provides sufficient information on the computer resources.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

Justification: There is no societal impact of the work performed.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA] .

    Justification: The paper poses no such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We cited the original owners of assets

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.

- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA] .

    Justification: The paper does not release new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA] .

    Justification: Our paper does not involve crowdsourcing nor research with human subjects

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA] .

    Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.