Supplementary Materials for Student Collaboration Improves Self-Supervised Learning: Dual-Loss Adaptive Masked Autoencoder for Multiplexed Immunofluorescence Brain Images Analysis

Anonymous submission

In this supplementary material, we present:

1. Related works: Section 1
2. Psuedo-code for DAMA: Algorithm 1
3. Noisy datasets experiments: Table 1
4. Visualization segmentation results: Fig. 1
5. Precision-Recall Curves: Fig 2, 3, 4, 5
6. Example of multiplexed brain cell images Fig. 6
7. Implementation details

1 Related Works

1.1 Self-Supervised Learning.

Recently, self-supervised learning (SSL) has exhibited a very successful approach in computer vision (Chen et al. 2020; Grill et al. 2020; He et al. 2020; Caron et al. 2021; He et al. 2021; Bao, Dong, and Wei 2021). However, choosing the right SSL learning algorithm is not always straightforward (Ericsson, Gouk, and Hospedales 2021). For example, one of the characteristics of multiplexed biomedical data is that the context also conveys crucial information about the cell. Learning self-supervised signals as multi-view augmented images with contrastive (Chen et al. 2020; Grill et al. 2020; He et al. 2020; Tsai et al. 2020; Lin et al. 2021), redundancy reduction (Zbontar et al. 2021) or self-distillation (Caron et al. 2021) objective would discard or unfocus on these context information. As an example, DINO (Caron et al. 2021) visualizes the attention maps of Vision Transformers (ViT) (Dosovitskiy et al. 2020) after training, whose main focus is on the interesting objects and leaves contextual information unattended. MAE (He et al. 2021) and SimMIM (Xie et al. 2021) learn to reconstruct missing image patches from uncorrupted patches. Similarly, Data2Vec (Baevski et al. 2022) regresses the unmasked patches from masked patches in feature level. However, due to the high random masking ratio, MAE, SimMIM, and Data2Vec would not guarantee to focus on the context information in each iteration. MoCo-v3 (Chen, Xie, and He 2021) learns to increase the mutual information of two augmented views of the same image, e.g., $I(X_1, X_2)$, due to the effect of augmentation transformations, MoCo-v3 would capture the cell body information and abandon the surrounding context information which distinct for each cell. Alternatively, based on ViT (Dosovitskiy et al. 2020) framework, we optimize the objective function on both pixel-level reconstruction (He et al. 2021; Xie et al. 2021; Bao, Dong, and Wei 2021) and features-level regression (Baevski et al. 2022) to predict the content of masked regions. By doing so, the algorithm will concentrate on invariant features and the entire image.

1.2 Masked Image Modeling (MIM).

Recent works built upon Vision Transformer (ViT) (Dosovitskiy et al. 2020) framework, such as BeiT (Bao, Dong, and Wei 2021), MAE (He et al. 2021), SimMIM (Xie et al. 2021) have shown potential of MIM in learning representations. Similar to our work, these prior studies propose masking out a random subset of image patches and encourage reconstructing the original pixel, but our work differs in that we also introduce regress feature representations of multiple ViT blocks (Baevski et al. 2022). On the other hand, our method is also distinct from Data2Vec (Baevski et al. 2022) as they take the masked and unmasked patches as input and predict features produced from uncorrupted input. We, however, apply only to the visible patches and predict the feature also produced from the visible patches of the second network, i.e., teacher or momentum network. Another point of separating our work from others is that we introduce an adaptive masking strategy that can learn better representation and boost fine-tune performance.

1.3 Self-Supervised Learning on Biomedical Data

Available SSL methods are usually applied to specific applications with less novelty contribution in the biomedical field. For instance, (Vicar et al. 2021) reconstructs distorted input to better representations of quantitative phase image cell segmentation, (Dmitrenko, Masiero, and Zamboni 2021) pre-trains 1M cancer cell images with convolutional autoencoder to classify the drug effects. Miscell (Shen et al. 2021) utilizes contrastive learning for mining gene information from single-cell transcriptomes. In addition, there are very few papers that study multiplexed biomedical data. While application-centric studies are acceptable, the theoretical analysis for adopting a specific SSL method is often unclear, missing out on potential approaches. In contrast, this study aims to bridge the gap between biomedical applications and theoretical motivation; and apply it to multiplexed biomedical data, e.g., brain cell data.

2 Noisy datasets

Real-30k and Real-170k Brain Cell Dataset. Augmentation could generate unlimited data. However, the underline structure of data is likely to remain the same. To exam our method on noisy data, we first cropped the large image into many 1000×1000 images and performed morphological transformations, i.e., erosion. These images were then applied watershed segmentation to identify the cells' location. From cells' center locations, we cropped with the size of 100×100 to get the images. To be compatible with the manually collected set, we collected 30k random cell images regardless of the cell type as the second training set, called *Noisy-30k* brain cells dataset. In addition, we further constructed another *Noisy-170k* brain cells dataset as a *large-scale* dataset.

Table 1 presents the comparisons of DAMA and other methods on *Noisy-30k* and *Noisy-170k*. MoCo-v3 has the best performance compared to others. This suggests that a small pretraining set has a negative impact on MoCo-v3. The results also indicate larger pretraining data and longer training time. MoCo-v3 can outperform other methods. This is expected since MoCo-v3 learns to increase the mutual information of two augmented views of the same image, e.g., $I(X_1, X_2)$, capturing better cell body information that is invariant across the dataset. In addition, the classification is considered not a rigorous task by utilizing the one-to-one cdence between cell types and biomarkers. However, this is also a downside since MoCo-v3 would abandon other critical information, e.g., contextual information. Data2Vec does not produce good results since it only learns to regress from low dimension features. On the other hand, MAE and DAMA have better results and are comparable to those from the original dataset.

Our DAMA achieves competitive results better than MAE and Data2Vec and is more stable on three brain cell datasets. MoCo-v3 is influenced by a small pretraining dataset but improves on a large dataset.

2.1 Precision-Recall Curves

We present the overall-all-all precision recall curves for bounding box and segment mask in Fig. 2 and 3. DAMA's results are come from imperfect localization *Loc* and background confusions *BG*. Regarding other methods, the amount of *Loc* and *BG* errors are higher than those of DAMA. Note that, we has only one class, i.e., segment cell body from background regardless its type.

The precision recall curves at different IoU threshold for bounding box and segment mask are shown in Fig. 4 and 5. For both detection and segmentation, DAMA has the best scores at the IoU from $0.1 : 0.75$ and are competitive at $0.8 : 0.9$.

2.2 Implementation Details

We implemented DAMA using Pytorch. Unless stated otherwise, we trained on ViT-Base used Adam optimizer (Kingma and Ba 2014) with base learning rate of 0.00015 (Chen, Xie, and He 2021), batch size of 512, image size 128×128×7, ViT patch size 16. Regarding state-of-the-arts implementation, we take the official released code (Chen, Xie, and He 2021; He et al. 2021; Touvron et al. 2021) and conduct pre-training with our biomedical data, except for Data2Vec (Baevski et al. 2022). We also use ViT-Base framework and similar parameters as above for these experiments. We report results of our DAMA and MAE (He et al. 2021) with masking ratios 80% and 60% for Data2Vec (Baevski et al. 2022). Training epochs and training times are listed along with the methods in result tables. All experiments were done on 4 GPUs of V100 32GB. Pre-training or finetune experiments of different methods on the same dataset have the same random seed.

Algorithm 1: Pytorch-like Adaptive Masking Pseudocode

```
def adaptive_mask(m1, loss, mask_ratio, overlap_ratio):
    # mask_ratio: masking ratio in [0, 1]
    # overlap_ratio: masks overlapping ratio between 2 inputs in [0, 1]
    # m1, m2: binary mask of 2 inputs where 0: unmasked and 1: masked; size[N, L]
    # loss: patch reconstruction losses; size[N, L]
    # N: batch size
    # L: total number of patches in images

    len_keep = int(L * (1 - mask_ratio))
    loss_len = int(L - len_keep * 2)
    overlap_len = int(len_keep * overlap_ratio)

    # get ids of high loss patches
    loss = loss * m1 # discard losses of unmasked patches in m1
    loss_sorted = argsort(loss)
    loss_take_ids = loss_sorted[:, -(loss_len + overlap_len):]

    # m1(1) becomes m2(0) and m1(0) becomes m2(1)
    m2 = where(m1 == 1, 0, 1)

    # assign ids of high loss patches to m2 as masked patches
    m2[arange(m2.shape[0])[:, None], loss_take_ids] = 1

    # overlap of unmasked patches of m1 and m2
    m1_ids = argsort(m1)
    m1_ids = m1_ids[:, :overlap_len]
    m2[:, m1_ids] = 0
return m2
```

Folds	0	1	2	3	4	5	6	7	8	9	Avg. ↑	Err. ↓
Random init.	91.75	91.19	92.75	92.69	92.56	92.31	91.44	91.06	93	91.06	91.98(+0.00)	8.02
Noisy-30k dataset												
Data2vec 800 (4h)	93.69	93	93.31	94.06	93.56	94.19	93.38	93.31	93.81	93.5	93.58(+1.60)	6.42
Data2vec 1600 (8h)	91.5	90.69	92.81	92.38	92.62	92.62	91.56	91.94	92.19	91.5	91.98(+0.00)	8.02
MOCO-v3 500 (6h)	95	94.12	95.81	96	95.75	95.19	95.12	94.44	95.5	95.19	**95.21(+3.23)**	**4.79**
MOCO-v3 1000 (12h)	94.44	93.62	94.38	95.19	94.69	95.25	94.12	94.38	95.25	94.31	94.56(+2.58)	5.44
MAE 800 (4h)	94.81	94.44	94.56	94.81	94	94	94.38	94	94.88	93.69	94.35(+2.37)	5.65
MAE 1600 (8h)	94.38	94.19	95.12	95.19	94.44	93.94	94.12	94.19	95.44	93.94	94.49(+2.51)	5.51
DAMA 500 (5h)	95	94.44	95.75	95.69	94.69	94.69	95.69	94.69	95.25	94.81	<u>95.07(+3.09)</u>	<u>4.93</u>
DAMA 1000 (10h)	94.5	94.06	95.75	95.44	94.69	94.44	94.44	94.38	95.25	94.25	94.72(+2.74)	5.28
Noisy-170k dataset												
Data2vec 800 (29h)	92.25	91.06	92.5	92.69	91.94	91.56	92.88	91.5	92.25	91.31	91.99(+0.01)	8.01
MOCO-v3 500 (48h)	95.38	94.56	95.94	95.94	95.81	95.38	95.62	95.25	95.81	95.56	**95.52(+3.54)**	**4.48**
MAE 800 (34h)	94.81	93.5	95	94.38	94.88	93.69	93.94	93.81	94.88	93.69	94.25(+2.27)	5.75
DAMA 500 (35h)	94.88	94.12	95.62	95.31	95.25	95.12	94.81	94.62	95.12	94.44	<u>94.92(+2.94)</u>	<u>5.08</u>

Table 1: Comparisons of fine-tuning results with state-of-the-art SSL methods pretrained with on *Noisy dataset* and randomly initialized in accuracy and error rate. Our DAMA reports stable results over dataset settings compared with other state-of-the-arts. We report results of our DAMA and MAE (He et al. 2021) with masking ratios 80% and 60% for Data2Vec (Baevski et al. 2022). Training epochs and training times are listed along with the methods. **Bold** and <u>underlined</u> are the highest and second highest scores, respectively.

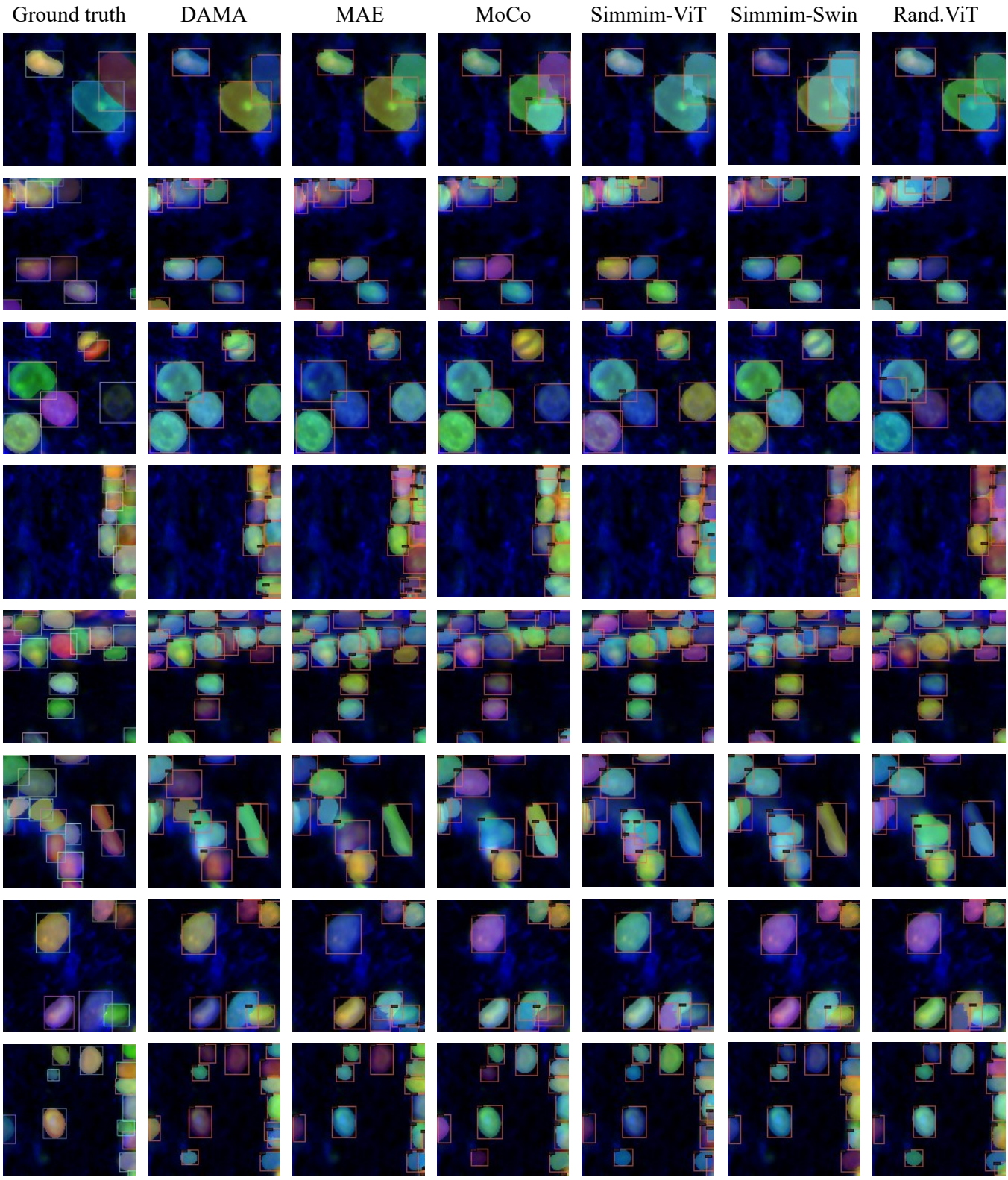
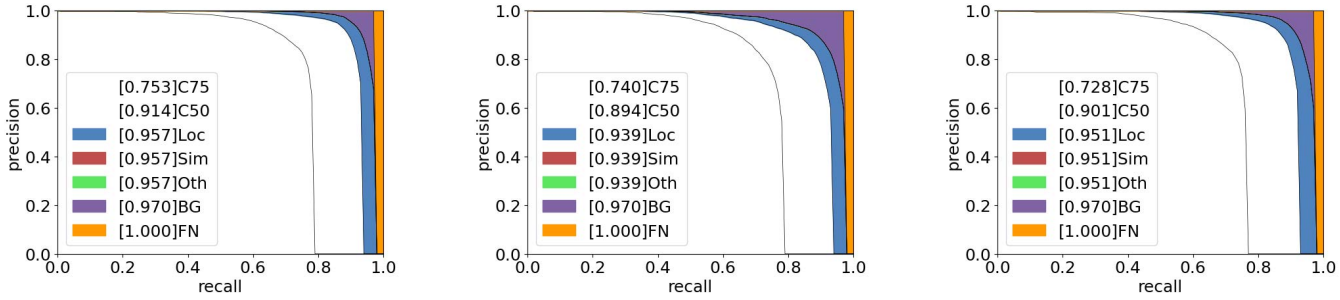


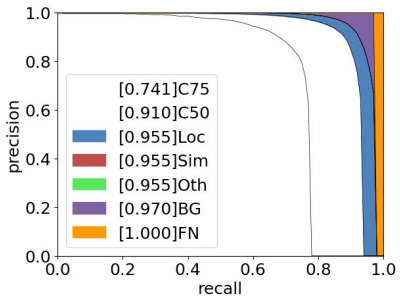
Figure 1: Visualization of segmentation results on validation set of DAMA and other methods at threshold IoU = 0.75. By focusing more on the contextual information, DAMA can detect and segment cells better where cells are dense and overlap.



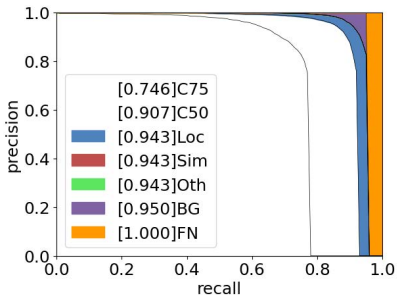
(a) DAMA

(b) MAE

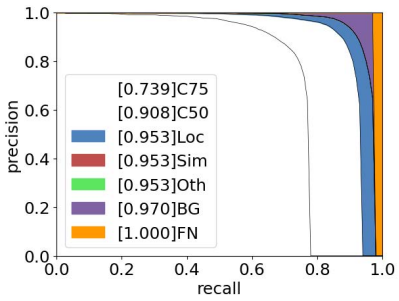
(c) MoCo



(d) SimMIM-ViT

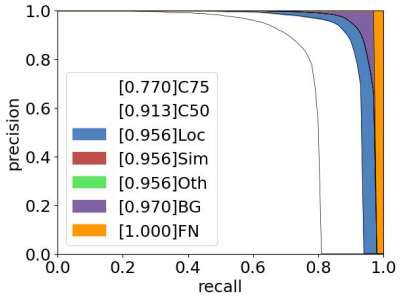


(e) SimMIM-Swin

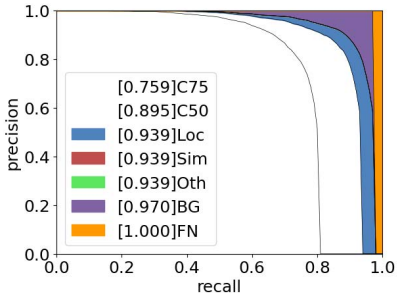


(f) Random init. ViT

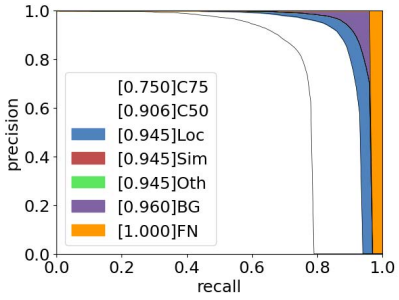
Figure 2: Bounding box error analysis: overall-all-all Precision-Recall curves of DAMA and other SSL methods. We follow (Hoiem, Chodpathumwan, and Dai 2012) and COCO evaluation analysis to produce plots. See *https://cocodataset.org/#detection-eval* for details.



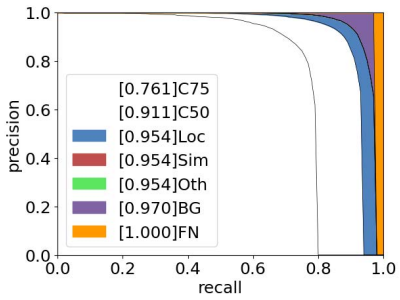
(a) DAMA



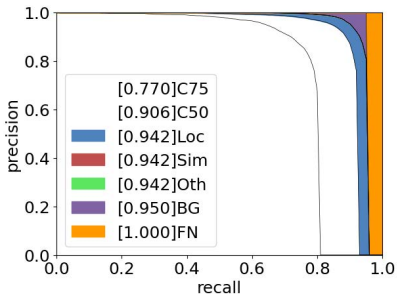
(b) MAE



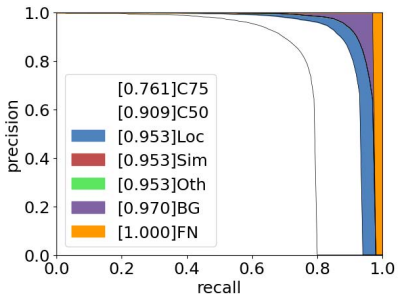
(c) MoCo



(d) SimMIM-ViT

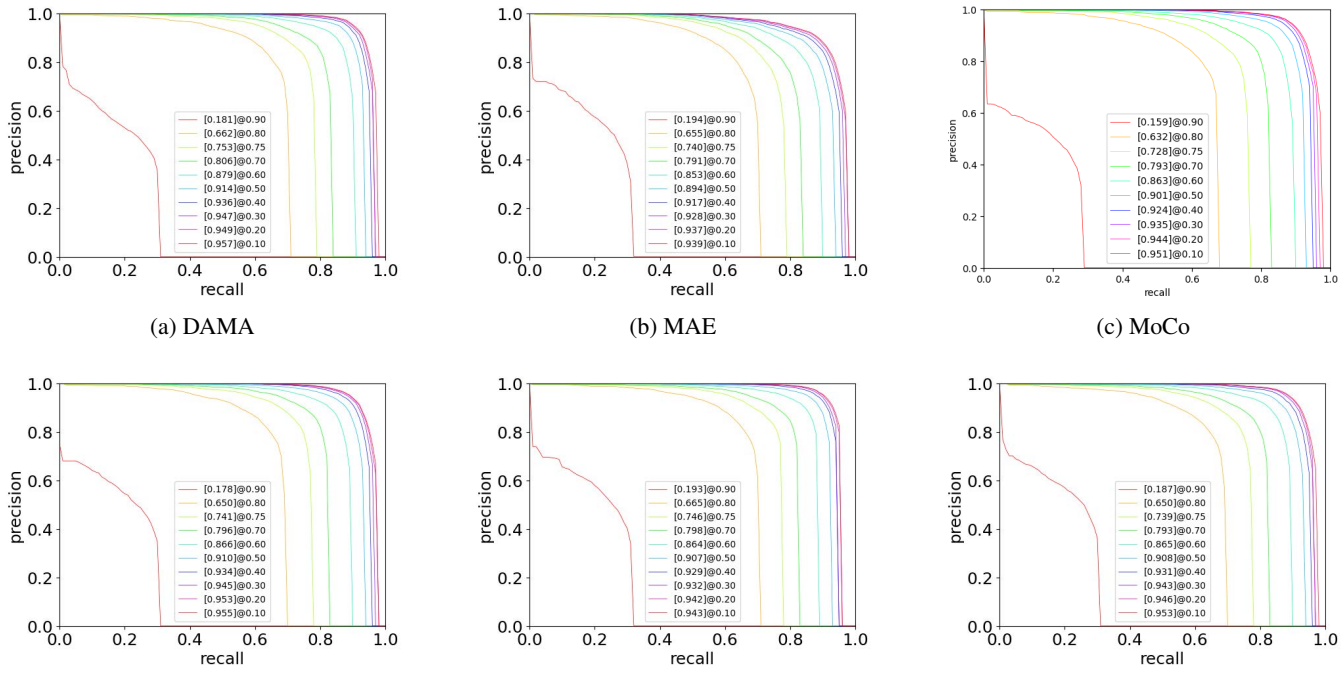


(e) SimMIM-Swin



(f) Random init. ViT

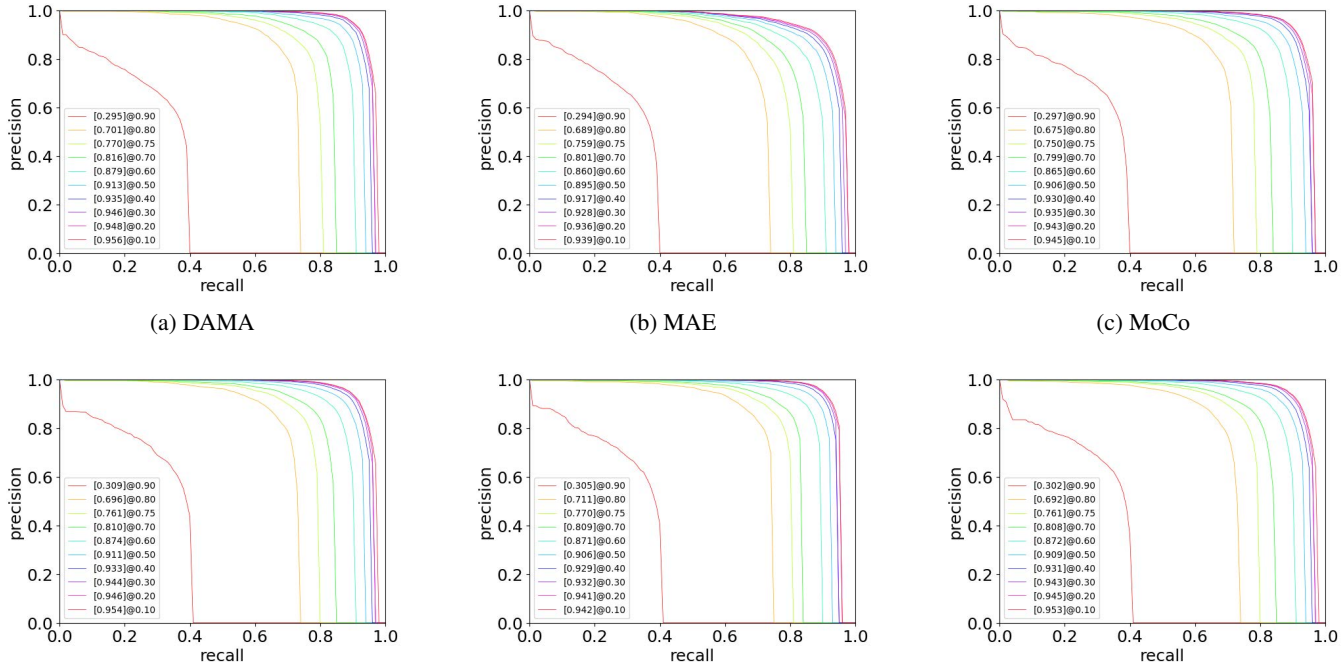
Figure 3: Segmentation mask error analysis: overall-all-all Precision-Recall curves of DAMA and other SSL methods. We follow (Hoiem, Chodpathumwan, and Dai 2012) and COCO evaluation analysis to produce plots. See *https://cocodataset.org/#detection-eval for details.*



(a) DAMA (b) MAE (c) MoCo

(d) SimMIM-ViT (e) SimMIM-Swin (f) Random init. ViT

Figure 4: Bounding box Precision-Recall curves of DAMA and other SSL methods at different IoU thresholds. DAMA has the best scores at the IoU from $0.1 : 0.75$ and are competitive numbers at $0.8 : 0.9$.



(a) DAMA (b) MAE (c) MoCo

(d) SimMIM-ViT (e) SimMIM-Swin (f) Random init. ViT

Figure 5: Segmentation mask Precision-Recall curves of DAMA and other SSL methods at different IoU thresholds. DAMA has the best scores at the IoU from $0.1 : 0.75$ and are competitive numbers at $0.8 : 0.9$.

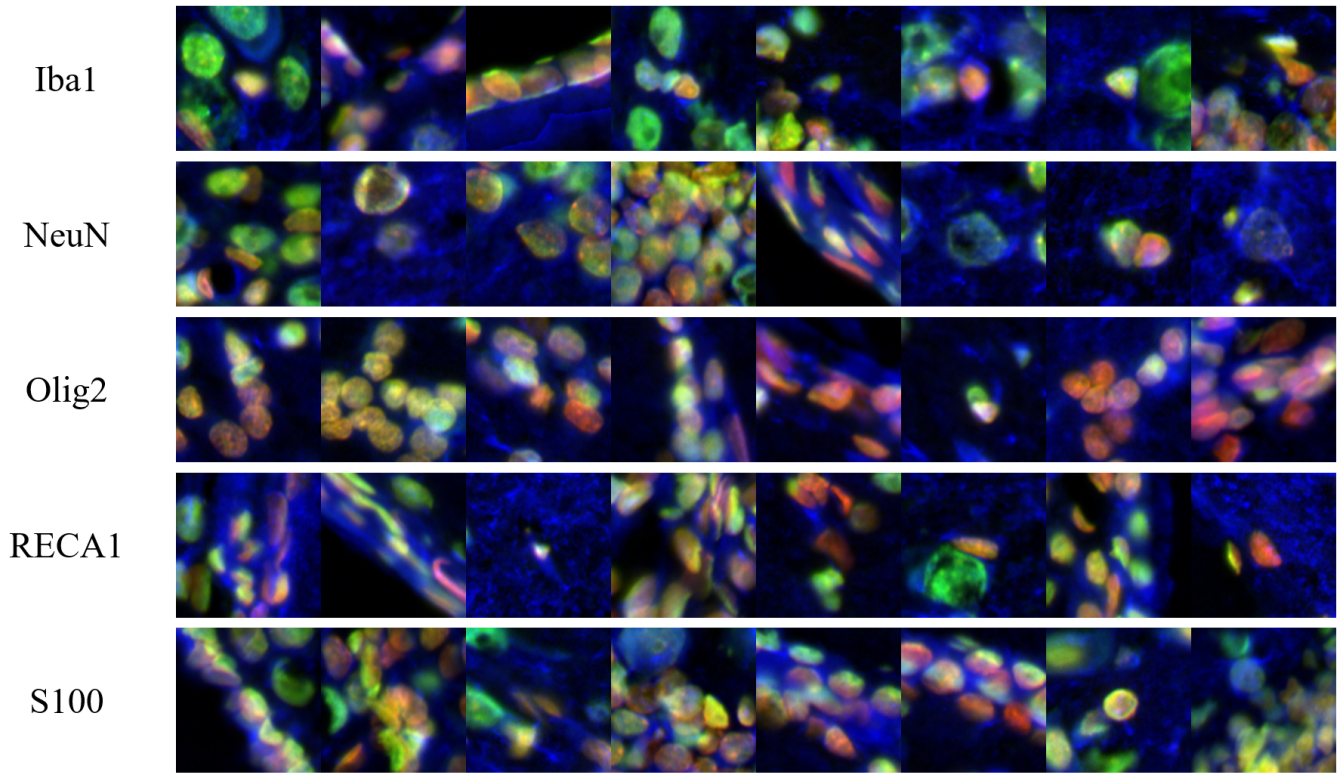


Figure 6: Example of five cell types: microglia, neurons, oligodendrocytes, endothelial, and astrocytes correspond to five biomarkers: Iba1, NeuN, Olig2, RECA1, S100.

Config	Value
image size	128×128×7
patch size	16×16
batch size	512
epochs	500
optimizer	Adam
base learning rate	1.5e-04
min learning rate	0
weight decay	0.05
learning rate schedule	cosine decay
warmup epochs	40
augmentation	RandomResizedCrop
K-blocks/β	6/2

Config	Value
image size	128×128×7
patch size	16×16
batch size	512
epochs	150
optimizer	Adam
Base learning rate	1e-02
min learning rate	1e-5
weight decay	0.05
learning rate schedule	cosine decay
warmup epochs	5
augmentation	RandomResizedCrop
droppath/reprob/mixup/cutmix	0.1/0.25/0.8/1.0

Table 2: Pre-training (left) and fine-tune (right) setting of our DAMA.

Config	Value
image size	128×128×7
patch size	16×16
batch size	512
epochs	500
optimizer	Adam
base learning rate	1.5e-04
min learning rate	0
weight decay	0.05
learning rate schedule	cosine decay
warmup epochs	40
augmentation	RandomResizedCrop

Config	Value
image size	128×128×7
patch size	16×16
batch size	512
epochs	150
optimizer	Adam
Base learning rate	1e-02
min learning rate	1e-5
weight decay	0.05
learning rate schedule	cosine decay
warmup epochs	5
augmentation	RandomResizedCrop
droppath/reprob/mixup/cutmix	0.1/0.25/0.8/1.0

Table 3: Pre-training (left) and fine-tune (right) setting of MAE (He et al. 2021).

Config	Value
image size	128×128×7
patch size	16×16
batch size	512
epochs	800/1600
optimizer	Adam
base learning rate	1.5e-04
min learning rate	0
weight decay	0.05
learning rate schedule	cosine decay
warmup epochs	40
augmentation	RandomResizedCrop
K-blocks/ β	6/2

Config	Value
image size	128×128×7
patch size	16×16
batch size	512
epochs	150
optimizer	Adam
Base learning rate	1e-02
min learning rate	1e-5
weight decay	0.05
learning rate schedule	cosine decay
warmup epochs	5
augmentation	RandomResizedCrop
droppath/reprob/mixup/cutmix	0.1/0.25/0.8/1.0

Table 4: Pre-training (left) and fine-tune (right) setting of Data2Vec (Baevski et al. 2022).

Config	Value
image size	128×128×7
patch size	16×16
batch size	512
epochs	500
optimizer	Adam
learning rate	1.5e-04
min learning rate	0
weight decay	0.1
learning rate schedule	cosine decay
warmup epochs	50
augmentation	RandomResizedCrop
stop-grad-conv1/moco-m-cos/moco-t	True/True/0.2

Config	Value
image size	128×128×7
patch size	16×16
batch size	512
epochs	150
optimizer	Adam
Base learning rate	1e-02
min learning rate	1e-5
weight decay	0.05
learning rate schedule	cosine decay
warmup epochs	5
augmentation	RandomResizedCrop
droppath/reprob/mixup/cutmix	0.1/0.25/0.8/1.0

Table 5: Pre-training (left) and fine-tune (right) setting of MoCo-v3 (Chen, Xie, and He 2021).

Config	Value
image size	128×128×7
patch size	16×16
batch size	512
epochs	300
optimizer	Adam
learning rate	1.5e-04
min learning rate	0
weight decay	0.1
learning rate schedule	cosine decay
warmup epochs	40
augmentation	RandomResizedCrop
droppath/reprob/mixup/cutmix	0.1/0.25/0.8/1.0

Table 6: Setting of random initialied experiment.

Config	Value
image size	224×224×3
patch size	16×16
batch size	4096
epochs	500
optimizer	Adam
base learning rate	1.5e-04
min learning rate	0
weight decay	0.05
learning rate schedule	cosine decay
warmup epochs	40
augmentation	RandomResizedCrop
K-blocks/β	6/2
mask ratio	0.8

Config	Value
image size	224×224×3
patch size	16×16
batch size	1024
epochs	150
optimizer	Adam
Base learning rate	1e-02
min learning rate	1e-5
weight decay	0.05
learning rate schedule	cosine decay
warmup epochs	5
augmentation	RandomResizedCrop
droppath/reprob/mixup/cutmix	0.1/0.25/0.8/1.0

Table 7: Pre-training (left) and fine-tuning (right) setting of DAMA on ImageNet-1k.

References

Baevski, A.; Hsu, W.-N.; Xu, Q.; Babu, A.; Gu, J.; and Auli, M. 2022. data2vec: A General Framework for Self-supervised Learning in Speech, Vision and Language. *arXiv preprint arXiv:2202.03555*.

Bao, H.; Dong, L.; and Wei, F. 2021. Beit: Bert pre-training of image transformers. *arXiv preprint arXiv:2106.08254*.

Caron, M.; Touvron, H.; Misra, I.; Jégou, H.; Mairal, J.; Bojanowski, P.; and Joulin, A. 2021. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 9650–9660.

Chen, T.; Kornblith, S.; Norouzi, M.; and Hinton, G. 2020. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, 1597–1607. PMLR.

Chen, X.; Xie, S.; and He, K. 2021. An empirical study of training self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 9640–9649.

Dmitrenko, A.; Masiero, M. M.; and Zamboni, N. 2021. Self-supervised learning for analysis of temporal and morphological drug effects in cancer cell imaging data.

Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Ericsson, L.; Gouk, H.; and Hospedales, T. M. 2021. How well do self-supervised models transfer? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 5414–5423.

Grill, J.-B.; Strub, F.; Altché, F.; Tallec, C.; Richemond, P.; Buchatskaya, E.; Doersch, C.; Avila Pires, B.; Guo, Z.; Gheshlaghi Azar, M.; et al. 2020. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in Neural Information Processing Systems*, 33: 21271–21284.

He, K.; Chen, X.; Xie, S.; Li, Y.; Dollár, P.; and Girshick, R. 2021. Masked Autoencoders Are Scalable Vision Learners. *arXiv:2111.06377*.

He, K.; Fan, H.; Wu, Y.; Xie, S.; and Girshick, R. 2020. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 9729–9738.

Hoiem, D.; Chodpathumwan, Y.; and Dai, Q. 2012. Diagnosing error in object detectors. In *European conference on computer vision*, 340–353. Springer.

Kingma, D. P.; and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Lin, Y.; Gou, Y.; Liu, Z.; Li, B.; Lv, J.; and Peng, X. 2021. COMPLETER: Incomplete multi-view clustering via contrastive prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11174–11183.

Shen, H.; Li, Y.; Feng, M.; Shen, X.; Wu, D.; Zhang, C.; Yang, Y.; Yang, M.; Hu, J.; Liu, J.; et al. 2021. Miscell: An efficient self-supervised learning approach for dissecting single-cell transcriptome. *Iscience*, 24(11): 103200.

Touvron, H.; Cord, M.; Douze, M.; Massa, F.; Sablayrolles, A.; and Jégou, H. 2021. Training data-efficient image transformers & distillation through attention. In *International Conference on Machine Learning*, 10347–10357. PMLR.

Tsai, Y.-H. H.; Wu, Y.; Salakhutdinov, R.; and Morency, L.-P. 2020. Self-supervised learning from a multi-view perspective. *International Conference on Learning Representations*.

Vicar, T.; Chmelik, J.; Jakubicek, R.; Chmelikova, L.; Gumulec, J.; Balvan, J.; Provaznik, I.; and Kolar, R. 2021. Self-supervised pretraining for transferable quantitative phase image cell segmentation. *Biomedical optics express*, 12(10): 6514–6528.

Xie, Z.; Zhang, Z.; Cao, Y.; Lin, Y.; Bao, J.; Yao, Z.; Dai, Q.; and Hu, H. 2021. Simmim: A simple framework for masked image modeling. *arXiv preprint arXiv:2111.09886*.

Zbontar, J.; Jing, L.; Misra, I.; LeCun, Y.; and Deny, S. 2021. Barlow twins: Self-supervised learning via redundancy reduction. In *International Conference on Machine Learning*, 12310–12320. PMLR.