Student Collaboration Improves Self-Supervised Learning: Dual-Loss Adaptive Masked Autoencoder for Brain Cell Image Analysis

Son T. Ly\textsuperscript{1}, Bai Lin\textsuperscript{1}, Hung Q. Vo\textsuperscript{1}, Dragan Maric\textsuperscript{2}, Badri Roysam\textsuperscript{1}, and Hien V. Nguyen\textsuperscript{1}

\textsuperscript{1} University of Houston, Dept. of Electrical and Computer Engineering 
\textsuperscript{2} National Institute of Neurological Disorders and Stroke, Bethesda

\{stly,lbai,hqvo2\}@uh.edu, \{broysam,hvnguy35\}@central.uh.edu

Abstract. Self-supervised learning leverages the underlying data structure as the source of the supervisory signal without the need for human annotation effort. This approach offers a practical solution to learning with a large amount of biomedical data and limited annotation. Unlike other studies exploiting data via multi-view (e.g., augmented images), this study presents a self-supervised Dual-Loss Adaptive Masked Autoencoder (DAMA) algorithm established from the viewpoint of the information theory. Specifically, our objective function maximizes the mutual information by minimizing the conditional entropy in pixel-level reconstruction and feature-level regression. We further introduce an adaptive mask sampling strategy to maximize mutual information. We conduct extensive experiments on brain cell images to validate the proposed method. DAMA significantly outperforms both state-of-the-art self-supervised and supervised methods on brain cells data and demonstrates competitive result on ImageNet-1k. Code: https://github.com/hula-ai/DAMA

Keywords: self-supervised learning, biomedical imaging application, brain cell classification.

1 Introduction

Microscopic brain image analysis is critical for medical diagnosis, and drug discovery \cite{21}. While collecting large amounts of brain imaging data using high-resolution multiplex microscopes is efficient, annotating these images is time-consuming and labor-intensive. Each brain slice consists of several hundred thousand cells and dozens of cell types. Labeling these images requires highly skilled biology experts. For these reasons, the number of labels for this application is often limited, while the amount of unannotated data is enormous. Self-supervised learning (SSL) offers a practical solution to this situation.

SSL methods have achieved impressive performance in natural language processing \cite{23,24,5,11}, speech processing \cite{22,25,3}, and computer vision \cite{7,17,6,112}.
SSL aims to learn powerful data representations useful for the downstream tasks by using pretext tasks automatically created without human supervision. Hence, this approach is ideal for biomedical applications with massive data and limited supervised information.

A few recent works have applied SSL to cell data \cite{12,32,26}. For instance, \cite{32} reconstruct distorted input to better representations of quantitative phase image cell segmentation, \cite{12} pre-trains 1M cancer cell images with convolutional autoencoder to classify the drug effects. Miscell \cite{26} utilizes contrastive learning for mining gene information from single-cell transcriptomes. None of these works studies SSL for multiplex brain cell analysis. In addition, existing works mainly focus on the applications, while the theoretical analysis for adopting a specific SSL method is often unclear. In contrast, this study tries to bridge the gap between biomedical applications and theoretical motivation. We propose a novel SSL framework that optimizes both pixel \cite{16,4,34}, and feature-level losses \cite{2,19,30} for brain cell image analysis as shown in Fig. 1. Our dual loss is motivated by information theory and the observation that the context around cells is useful for classifying them correctly. Our method maximizes the mutual information between masked inputs \cite{16,42} and self-supervised signals. Specifically, we first mask the original input and then map it to feature space wherein its repre-
sentations learn to reconstruct the original unmasked input. Simultaneously, the exact representations also regress to the representations of a different masked version of the original input. Synchronously learning pixel reconstruction and feature regression increases the consistency of different masked images from the same input. In addition, from the information standpoint and our observation, we further propose an adaptive masking strategy to enforce the networks to learn better representations.

We perform SSL pre-training and then fine-tune our method on cell types classification task with our collected biomedical data, i.e., brain cells image and on ImageNet-1k [10]. Experimental results indicate that our method outperforms both SSL state-of-the-art models and from-scratch training. Ablation analysis shows the effectiveness of our method in learning good representation. The main contributions of our paper are as follows:

1. We present a novel SSL method for biomedical data analysis, i.e., brain cell, motivated from the information theory perspective. Our method achieved superior performance compared to state-of-the-art SSL methods and supervised. Extensive experiments on brain cells dataset and ImageNet-1k are conducted to support the proposed method.
2. We also propose an adaptive mask sampling strategy that considerably influences learning good representation. This could be the first adaptive masking method for self-supervised learning to the best of our knowledge.

2 Related Works

**Self-Supervised Learning.** Recently, self-supervised learning (SSL) has exhibited a very successful approach in computer vision [7,15,17,6,16,4]. Contrastive learning [7,17,15] distinguishes the similarity and dissimilarity, or similarity only [6] (self-distillation) of multi-views; thus, it depends on data augmentation to generate different views. However, choosing the right SSL learning algorithm is not always straightforward. For example, one of the characteristics of biomedical data is that the context also conveys crucial information about the cell. Learning self-supervised signals as multi-view augmented images with contrastive [7,15,17,30,19], redundancy reduction [35] or self-distillation [6] objective would discard or unfocus on these context information. As an example, DINO [6] visualizes the attention maps of Vision Transformers (ViT) [13] after training, whose main focus is on the interesting objects and leaves contextual information unattended. Alternatively, based on ViT [13] framework, we optimize the objective function on both pixel-level reconstruction [16,34,4] and features-level regression [2] to predict the content of masked regions. By doing so, the algorithm will concentrate not only on invariant features but also on the entire image.

**Masked Image Modeling (MIM).** Recent works built upon Vision Transformer (ViT) [13] framework, such as BeiT [4], MAE [16], SimMIM [34] 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 [2]. On the other hand, our method is also distinct from Data2Vec [2] 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.

Self-Supervised Learning on Biomedical Data Parallel to the solely theoretical aspect, research in SSL also focuses on practical applications. Available SSL methods are usually applied to specific applications with less novelty contribution in the biomedical field. For instance, [32] reconstruct distorted input to better representations of quantitative phase image cell segmentation. [12] pre-trains 1M cancer cell images with convolutional autoencoder to classify the drug effects. Miscell [26] utilizes contrastive learning for mining gene information from single-cell transcriptomes. 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.

3 Self-Supervised Learning from Information Theory Perspective

Notations. For the rest of this paper, we denote the input and self-supervised signal in general as \(X\) and \(S\), respectively. \(S\) can be the augmented image [7,17,6] or the target of image reconstruction [16,34,4]. The deterministic mapping function \(F\) maps the input \(X\) to its representations \(Z_X\), i.e. \(Z_X = F(X)\), and function \(G\) reconstructs the input as \(S = G(Z_X)\). Regarding the information, we coin the mutual information, entropy, and conditional entropy of variable \(A\) and \(B\) are \(I(A, B)\), \(H(A)\), and \(H(A|B)\), respectively.

As shown in Fig. 2, solid and dotted rectangles represent the information of input \(X\) and self-supervised signal \(S\), respectively. From the information theory perspective, the mutual information between the representation \(Z_X\) and \(S\), denoted as \(I(Z_X, S)\) (grey area), measures the amount of information obtained about one from the knowledge of the other. This mutual information can be expressed as the difference of two entropy terms:

\[
I(Z_X, S) = H(Z_X) - H(Z_X|S) = H(S) - H(S|Z_X)
\]  

(1)

In self-supervised learning context, one can directly maximize \(I(Z_X, S)\) like in Completer [19]. Alternatively, minimizing the conditional entropy \(H(S|Z_X)\)
Fig. 2. Overview of our pipeline in information perspective.

(green area) \[17,22\] would also encourage \(S\) to be fully determined by \(X\), indirectly maximizing \(I(Z_X, S)\) and minimizing the irrelevant information between \(X\) and \(S\). The Eq.(1) can be interpreted as \(I(Z_X, S)\) minimize the uncommon information between \(X\) and \(S\) \[30,35\]. Hence, if \(X\) and \(S\) are independent, then \(I(Z_X, S) = 0\), while if \(X\) and \(S\) are related, then \(I(Z_X, S)\) will greater than some lower bound. For this reason, \(S\) is usually the augmented images \[7,17,15,6,35,30,2\] or random masked images \[16,34,4,2\].

Augmentation could boost the performance of self-supervised learning algorithm \[7,15,17\]. From the Information Bottleneck principle \[28,27\], augmented images could enforce the encoder \(F\) to estimate invariant information \[33\]. However, augmentation is data-dependent, and finding the right transformation could be inconvenient in some cases. SimCLR \[7\] conducts a resource-consuming experiment with the combination of only two transformations to find the most favorable combination for ImageNet-1k \[10\]. Moreover, augmentation could remove contextual information, which is important for downstream biomedical tasks.

Motivated by \[30,19\], our method aims to minimize the conditional entropy \(H(S|Z_X)\). Fig. 2 provides an illustration of our method. While \[30\] employs forward-inverse predictive learning to boost the performance of contrastive objective, \[19\] benefits from dual prediction and contrastive learning for recovering missing views. We, however, take a fundamentally different approach by not targeting to optimize the contrastive function \[30,19\] but focusing on maximizing the mutual information between masked inputs and self-supervised signals at pixel-level reconstruction \[16,34\] and features-level regression \[2\]. In addition, while most existing works \[16,4,34,2\] utilize a random image masking strategy, our adaptive sampling method could further minimize the conditional entropy \(H(S|Z_X)\) and learn better representations. To the best of our knowledge, our method is the first to use adaptive image masking for self-supervised learning.

4 Dual-loss Adaptive Masked Autoencoder

Motivated from the information theory perspective, this section proposes a Dual-loss Adaptive Masked Autoencoder (DAMA) for self-supervised learning. Our method optimizes an objective function associated with information masking
at both pixel- and feature-level. As illustrated in Fig. 1, it consists of a dual objective function:

$$\mathcal{L}_{total} = \mathcal{L}_p + \alpha \mathcal{L}_f$$  \hspace{1cm} (2)$$

where $\mathcal{L}_p$ and $\mathcal{L}_f$ are the reconstruction pixel-level and regression feature-level, respectively. $\alpha$ is a non-negative constant. From the information theory perspective, our method optimizes both $I(Z_X, S)$ and $H(S|Z_X)$. In addition, we also present the adaptive masking strategy that has more promise than random masking in terms of performance and theory background. We first provide context information related to the method development and introduce theoretical details later. In our implementation, we fixed $\alpha = 1$ for all experiments.

DAMA is developed based on Vision Transformer (ViT) [13] framework. Given input image $x \in \mathbb{R}^{H \times W \times 3}$, we reshape it into small patches $(x^P)_i^{N}$, where $N = HW/P^2$ patches and $P$ is the resolution of each patch. We masked $m\%$ of the patches and denote them as $M = \{1,..., N\}^m$. Here, unlike BEiT [4] and Data2Vec [2] treat the masked patches and unmasked patches as input to ViT, i.e. $x^P = \{x^P_i : i \notin M\}^{N}_{i=1} \cup \{e_i^P : i \in M\}^{N}_{i=1}$, where $e_i^P$ is the learnable embedding replacing for masked patches, we feed only the unmasked patches $x^P_U = \{x^P_i : i \notin M\}^{N}_{i=1}$ which similar to MAE [16].

4.1 Pixel-level Reconstruction

Here, we present the theoretical background for the pixel-level loss $\mathcal{L}_p$ in Eq. (2). In the context of pixel reconstruction, we regard the self-supervised signal $S$ as the reconstruction target, i.e., original input, indicated as $S_X$ in Fig. 1.

According to Eq. (1), minimizing the conditional entropy $H(S|Z_X)$ (green area) would also encourage $S$ to be fully determined by $X$, indirectly maximize $I(Z_X, S)$, and minimize the irrelevant information between $X$ and $S$ [35,30]. To do so, the learned representation $Z_X$ is encouraged to reconstruct the self-supervised signal $S$ which lead to maximize the log conditional likelihood, by the chain rule: $-H(S|Z_X) = \mathbb{E}_{P_S, Z_X} [\log P(S|Z_X)]$. However, directly inferring $P(S|Z_X) = \frac{P(Z_X|S)P(S)}{P(Z_X)}$ would be intractable [1]. A common approach to approximate this objective is to define a variational distribution $Q(S|Z_X)$ and maximize the lower bound $\mathbb{E}_{P_S, Z_X} [\log Q(S|Z_X)]$ using variational information maximization technique [1], i.e.,

$$\mathcal{I}(Z_X, S) = H(S) - H(S|Z_X)$$

$$= \mathbb{E}_{P_S, Z_X} [\log P(S|Z_X)] + H(S)$$

$$= D_{KL}(P(S|Z_X) \parallel Q(S|Z_X)) + \mathbb{E}_{P_S, Z_X} [\log Q(S|Z_X)] + H(S)$$  \hspace{1cm} (3)$$

Such $Q(\cdot | \cdot)$ can be any type of Gaussian [14,30,19], Laplacian [37], categorical [9] distribution, or neural network [20,31]. We present the $Q(S|Z_X)$ as Gaussian distribution with $\sigma I$ as diagonal matrix, i.e., $N(S|G(Z_X), \sigma I)$ [14,30,19], where
where $Z$ suggests to maximize the log conditional likelihood $E$ patches of $X$ where $Z$ is the sample space of $S$. After specifying the mapping functions $G$, the maximizing $E_{P_S, Z_X} \left[ \log Q(S | Z_X) \right]$ objective functions in pixel-level is obtained as:

$$
\mathcal{L}_p = \min \sum_{i=1}^{n} \left[ E_{P_{S, Z_X}^m} \left\| G(Z_{X_i}^m) - S \right\|_2^2 \right], \quad i = 1, 2
$$

where $Z_{X_i}^m$ is the representation of masked input $X_i^m$, i.e. $Z_{X_i}^m = F(X_i^m)$, and $i = 1, 2$ represents for two branches of models as shown in Fig. 1. From the above objective function, we can notice that when $\mathcal{L}_p = 0$, i.e., $S$ can be fully determined by $Z_{X_i}^m$, mathematically, $H(S | Z_{X_i}^m) = 0$. One common situation is that the $G(F(\cdot))$ becomes the identical mapping $I$, and the network will learn nothing from this. For this reason, $X$ is usually the augmented images or random masking images from the same source to avoid the situation. Our DAMA employs the image masked autoencoder modeling approach similar to [16,4,34,2] instead of augmenting the input. To leverage the masking operation to contribute more than just generating random masks, we propose an adaptive masking strategy that can increase the mutual information $I(Z_X, S)$ and learn better representation whose details will be explained in section 4.3.

### 4.2 Feature-level Regression

To further encourage maximizing the mutual information $I(Z_X, S)$, DAMA also consist a feature-level regression objective $\mathcal{L}_f$ in Eq. 2. In the context of feature-level regression, we prefer self-supervised signal $S$ as the feature target produced by $F_Z(X_2^m)$ and indicate as $S_F$ in Fig. 1. Difference to Data2Vec [2] which predicts feature representations of the original uncorrupted input $X$ based on a masked view $X^m$ in a student-teacher setting, i.e. $F_Z : X^m \mapsto X$, where $F_Z$ is the mapping function, DAMA predicts feature representations of masked view $X_2^m$ based on masked view $X_1^m$, i.e. $F_Z : X_1^m \mapsto X_2^m$. Furthermore, the masked patches of $X_2^m$ are decided by adaptive sampling strategy.

Similar as pixel-level context, maximizing the mutual information $I(Z_X, S)$ suggests to maximize the log conditional likelihood $E_{P_S, Z_X} \left[ \log P(S | Z_X) \right]$. This lead to in introducing variational distribution $Q(S | Z_X)$ to maximize the lower bound $E_{P_S, Z_X} \left[ \log Q(S | Z_X) \right]$. Let $Q(S | Z_X)$ as Gaussian distribution with $\sigma I$ as diagonal matrix, i.e., $N(S | F_Z(Z_X), \sigma I)$. The objective functions is obtained as:

$$
\mathcal{L}_f = \min \mathbb{E}_{P_{S, Z_X}^m} \left\| F_Z(Z_{X_1}^m) - S \right\|_2^2
$$

where $Z_{X_1}^m$ is the representation of masked input $X_1^m$, i.e. $Z_{X_1}^m = F(X_1^m)$. Note that $F$ and $F_Z$ are two different mapping functions; refer to Fig. 1 to clarify. Given that our main interesting application is for the biomedical dataset, i.e., brain cell, where context is also crucial information, we adopt the smooth L1 loss presented in [2]. Hence, the Eq.(5) becomes:

$$
\mathcal{L}_f = \begin{cases} 
\frac{1}{2} (F(Z_{X_1}^m) - S)^2 / \beta, & \left| F(Z_{X_1}^m) - S \right| \leq \beta \\
\left| F(Z_{X_1}^m) - S \right| - \frac{1}{2} \beta, & otherwise
\end{cases}
$$
where $\beta$ is the smoothing from L2 to L1 loss term and depends on the difference between $F_Z(Z_{X^m_1})$ and $S$. In addition, the self-supervised signal $S$ is taken from the last $K$ blocks of the second branch of models before normalization to each block and then averaging similarly as in Data2Vec [2]. In our implementation, we set $K = 6$, $\beta = 2$ for all experiments.

### 4.3 Adaptive Masking Strategy

Unlike other works [16,34,24], we propose an adaptive masking strategy to not only produce masked images $X^m_2$ but also increase the mutual information $I(Z_X, S)$ and learn better representations. The method is originated from our observation for the theoretical background presented in the pixel-level reconstruction section 4.1. The patches with the highest loss indicate the lowest mutual information $I(Z_X^m, S)$.

As summarized in Algorithm 1, the proposed strategy takes the random binary mask of $X^m_1$ and the patch reconstruction loss in Eq.(4) as inputs. It selects the patches with the highest loss, which indicate the lowest mutual information $I(Z_X^m, S)$ as masked patches for $X^m_2$. Regarding the unmasked patches in $X^m_1$, based on the overlap ratio, some of them will become the unmasked patches, and the rest will serve as masked patches in $X^m_2$. The overlap ratio is fixed at 50% for all experiments. This guarantees that the feature-pixel regression would not be too difficult to predict. One can think of the adaptive image masking strategy as a collaboration between two students, where the first student estimates the difficulties of reconstructing different patches, and the second student uses that information to select challenging patches to enhance the performance. Note that we develop DAMA upon ViT framework [13]. Hence, we compute the reconstruction loss patch-wise, and unmasked patches are not taken into account in computing loss [16,24,34].

### 5 Experiments

In this section, we validate DAMA algorithm on the brain cell dataset and compare its fine-tuning performance with supervised learning and state-of-the-art SSL approaches.

#### 5.1 Experimental Settings

**Aug-30k Brain Cell Dataset.** Cell classification is a fundamental building block in the whole-brain tissue profiling pipeline, an essential tool to understanding brain diseases and drug discovery. We manually collected 4000 cell images from 5 major cell types from rat brain tissue sections: neurons, astrocytes, oligodendrocytes, microglia, and endothelial. There are 800 cells for each cell type. Seven biomarkers are applied as the feature channels: DAPI, Histones, NeuN, S100, Olig 2, Iba1, and RECA1. DAPI and Histones are utilized to reveal the cells’ locations, while others biomarkers are for classifying specific cell types.
We treat DAPI, Histones, and others’ average as 3 RGB channels. Each cell is located in the center of a patch with the size of 100×100 pixels. No cell detection pre-processing has been applied to these images; thus, some patches may contain more than one cell, but only the central cell is what we are interested in, and all other cells are generally considered as the background for the cell type classification task, see the Fig. [A.1] for example cell images. We split each cell type image into 60%/40%, i.e., 2400/1600, to be the training/validation set and performed affine augmentation on the training set, resulting in a total of 30k images, called Aug-30k brain cells dataset. The transformations are: rotating, translating, flipping, scaling, and no colorizing transformation.

**Real-30k 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 cells’ location. From cells’ center locations, we cropped with the size of 100×100 to get the images. We totally collected 30k random cell images regardless of the cell type as the second training set, called Real-30k brain cells dataset.

**Implementation Setting.** We implemented DAMA using Pytorch, and all experiments were done on 4 GPUs of V100 32GB. Unless stated otherwise, we trained on ViT Tiny for 500 epochs and used Adam optimizer [18] with base learning rate of 0.00015 [8], batch size of 512, image size 128×128×3, ViT patch size 8.

Regarding state-of-the-arts implementation, we take the official released code [8][16][29] and conduct pre-training with our biomedical data, except for Data2Vec[2]. We also use ViT Tiny framework and similar parameters as above for these experiments. Regarding fine-tuning, our method, MAE [16], and Data2Vec [2] utilize the fine-tune code of MAE, while MoCo-v3 [8] are done with DeiT [29]. For supervised experiments, we trained the ViT Tiny with 300 epochs on modified MAE [16] code. We also tried with DeiT [29] but observed lower accuracy compare with using modified MAE [16] code. See Appendix A.1 for parameter details.

### 5.2 Comparisons with State-of-the-arts

**Brain cells data.** In Table 1 and 2 we compare the fine-tuning results of supervised ViT and self-supervised ViT models pre-trained on Aug-30k and Real-30k. Our DAMA exceeds state-of-the-arts self-supervised methods and scratch supervised with large margin on both datasets.

Regarding MAE [16] and Data2Vec [2], our DAMA is more accurate even with random masking setting. This is expected since DAMA is generally a combination of both. MAE [16] and DAMA were pre-trained on 500 epochs for three

---

3 Data2Vec code for computer vision has not been released.
Table 1. Comparisons of fine-tuning results with state-of-the-arts self-supervised and supervised methods. Our DAMA exceeds state-of-the-arts with a large margin on both datasets. We report results of our method and MAE [16] with masking ratios of range 70%/80%/90% and 60% for Data2Vec [2]. Subscripts indicate the pre-trained epochs. Superscripts indicate masking strategy.

| Methods                  | Pre-training Sets | Aug-30k          | Real-30k          |
|--------------------------|-------------------|------------------|------------------|
| Supervised ViT300        |                   |                  |                  |
| MoCo-v3500 [8]           |                   | 73.75            | 77.19            |
| MAE500 [16]              |                   | 66.69/65.31/65.31| 65.19/65.31/67.25|
| *Data2Vec800 [2]         |                   | 73.12            | 67.94            |
| *Data2Vec1600 [2]        |                   | 76.31            | 73.69            |
| Ours DAMA\textsuperscript{random}500 |       | 74.56/76.31/76.96| 74.31/76.38/75.81|
| Ours DAMA\textsuperscript{adaptive}500 |         | -/77.69/\textbf{78.19} | -/\textbf{78.12}/77.25 |

(*) We implemented Data2Vec by ourselves with random patch masking. 
(-) Insufficient GPUs memory.

different mask ratios (70%, 80%, and 90%). MAE reports high optimal masking ratios, ranging in 40%-80% on ImageNet-1K. For this dataset, MAE produces similar results for ratio in the range of 70%-90% except for 67.25% on Real-30k. Since Data2Vec [2] has similar approach to our feature-level regression, we implemented Data2Vec [2] ourselves with random patch masking ratio 60% and pre-trained with 800 and 1600 epochs as in [2]. Our DAMA outperforms both 800 and 1600 pre-training epochs of Data2Vec. Regarding pre-training time and memory consumption in Table 2, our numbers are doubled that of MAE [16] as DAMA uses another network to learn the second mask. Data2Vec [2] only leverages features for learning, leading to significantly less training time and memory.

Compared with MoCo-v3 [8], our DAMA surpasses in terms of performance. Our method reconstructs pixels and regresses features in contrast to MoCo-v3 that optimizes thought contrastive learning. MoCo-v3 is 1 hour faster and has less memory than our adaptive masking method. On the other hand, DAMA with the random masking setting has 4.5-5 hours of competitive training time. In addition, both of our results improve over MoCo-v3. Our DAMA and MoCo-v3 were both trained with 500 epochs. Note that MoCo-v3 [8] reported to achieve good results on ImageNet-1K for pre-training with only 300 epochs.

All the experiments in Table 1 are done with the optimal pre-training batch size 512. Under the constrain of computational GPUs resources, to fairly compare with state-of-the-arts, we report the result of batch size 512 for all methods in Table 1.

ImageNet-1k. To demonstrate the potential of DAMA on other natural image type, we present result on ImageNet-1k [10], see Table 3. Compared with other
**Table 2.** Comparisons of training time and memory (hour, MB). We report the values of our method and MAE [16] with masking ratios of range 70%/80%/90% and 60% for Data2Vec [2].

| Methods       | Time (h)         | GPUs Memory (MB)       |
|---------------|------------------|------------------------|
| Moco-v3500    | 5h52m            | 13235                  |
| MAE500        | 3h43m/3h39m/3h29m| 15525/15147/14819       |
| Data2Vec<sup>*</sup>500 | 4h15             | 6211                   |
| Data2Vec<sup>*</sup>1600 | 8h10             | 6211                   |
| Ours<sup>random</sup>500 | 4h45m/4h38m/4h31m | 16356/15942/15575       |
| Ours<sup>adaptive</sup>500 | -/6h51m/6h45m     | -/29715/29057           |

(*) We implemented Data2Vec by ourselves with random patch masking.
(-) Insufficient GPUs memory.

state-of-the-art algorithms, our DAMA is proven to be competitive while pre-trained with smaller epochs and without any ablation experiment for searching optimal hyper-parameters.

Due to the computational resource needed for training on such a large-scale dataset, we perform a single pre-training/fine-tuning experiment with a similar configuration as for training brain cells dataset, except for the model, image size, and pre-training/fine-tuning batch size as Vit-Base, 224 × 224, and 4096/1024, respectively. See Table A6 for parameter details.

**Table 3.** Comparisons results of DAMA and state-of-the-arts on ImageNet-1k with ViT-Base/16.

| Methods       | Pre-training epochs | Accuracy |
|---------------|---------------------|----------|
| Moco-v3*      | 600                 | 83.2     |
| BEiT          | 800                 | 83.4     |
| SimMIM        | 800                 | 83.8     |
| Data2Vec      | 800                 | 84.2     |
| DINO*         | 1600                | 83.6     |
| iBOT*         | 1600                | 84.0     |
| MAE           | 1600                | 83.6     |
| Ours DAMA     | 500                 | 83.17    |

(*) Results reported in [36].

### 5.3 Ablation Studies

In table 4 and Fig. 3, we ablate our DAMA with different masking strategies, model strategies, and masking ratios.
Table 4. Effect of sampling strategies, model settings, and masking ratios. All the experiments are trained on the Aug-30k dataset. Parameters are not presented in the table remain the same for all experiments. Overlapping means some unmasked patches in $X_1^m$ will also be unmasked in $X_2^m$. **Bold** are the highest score for each column, while **underlined** are highest score for each row.

| Sampling Strategy            | Model Strategy         | Masking Ratio (%) |
|------------------------------|------------------------|-------------------|
|                              |                        | 60    | 70    | 80    | 90    |
| Random with Overlapping      | student-teacher        | 74.31 | 74.56 | 76.31 | 76.96 |
| Adaptive without Overlapping | student-teacher        | 74.81 | 74.5  | 75.56 | 75.38 |
| Adaptive with Overlapping    | student-teacher        | 74.75 | 75.81 | 76.69 | 77.88 |
|                              | share-weights          | -     | -     | 77.12 | 76.00 |
|                              | student1-student2      | -     | -     | 77.69 | 78.19 |

(-) Insufficient GPUs memory.

**Masking Strategies.** We compare our DAMA with **adaptive** and **random** masking strategies and analyze how they affect the fine-tune results. The adaptive masking produces better accuracy in the same training condition with different masking ratios (70%-90%). A higher masking ratio achieves a better result.

*Overlapping*, a step in the proposed adaptive masking, means some unmasked patches in $X_1^m$ will also be unmasked in $X_2^m$. The overlapping ratio is set at 50% for all experiments. This parameter also affects performance. No overlapping between unmasked input $X_1^m$ and $X_2^m$ leads to the feature-level mutual information $I(Z_X, S)$ being more challenging to optimize $L_f$, resulting in lower fine-tuning results. As illustrated in Fig. 3, the loss curves for none overlapping experiments are higher than others. These results confirm the effect of adaptive masking and feature-level regression on the overall objective function.

**Model Settings.** In masking strategies ablation, we have verified the impact of **adaptive masking** via no overlapping between unmasked inputs $X_1^m$ and $X_2^m$ can downgrade the feature-level mutual information $I(Z_X, S)$. The effects of **adaptive masking** can be explained similarly under **model settings** scope. Recall that some unmasked patches in $X_1^m$ will also be unmasked in $X_2^m$, the two deterministic mapping functions $F : X^m \rightarrow Z^m$, and $F_Z : Z_1^m \rightarrow Z_2^m$. In the **student-teacher** setting, teacher network is an exponential moving average (EMA) on the student weights [17], and the update rule is $\theta_t \leftarrow \lambda \theta_t + (1 - \lambda) \theta_s$, where $\lambda$ follows cosine schedule from 0.996 to 1 during training [15][17][6].

Regarding three model settings, the **student-teacher** is less effective than **share-weights** (student1-student1) and **student1-student2** setting, shown in Table 4 and Fig. 3. The reason could be that the **teacher** in the first setting are the **exponential moving average** [6], while those of in second and third setting are the **identical** and the independent **student2** network from **student1**, respectively.
Table 5. Fine-tuning results of Network1 and Network2 in different model settings, masking ratio 80%.

| Sampling strategy            | Model settings | Network1 | Network2 |
|------------------------------|----------------|----------|----------|
| Adaptive w/o overlapping     | student-teacher| 75.56    | 75.06    |
| Adaptive with overlapping    | student-teacher| 76.69    | 76.19    |
|                              | share-weights  | 77.12    | 76.88    |
|                              | student1-student2| 77.69    | 77.69    |

Specifically, in student-teacher setting, since the teacher is the exponential moving average [6] of student, the feature-level representations $Z^m_2$ and $Z^m_1$ would be very inconsistent comparing to those of share-weights and student1-student2 setting. Regarding the student1-student2 setting that consists of two independent networks which will have more “freedom” in optimizing objective functions. In other words, the feature-level loss $L_f$ for student-teacher would be the highest among three settings. In contrast, those of for share-weights and student1-student2 would be lower. These hypotheses are certified in the loss training curves plots in Fig. 3.

In addition, this also influences the fine-tuning results between two networks in all three model settings pre-trained with masking ratio 80%, as shown in Table 5. While the teacher in the first setting have lower performances than its student, the student2 who has more independent in optimizing objective functions offer similar results to student1. In this figure, all the experiments are done with the same parameters, except for the model setting.

Masking Ratios. According to Table 4, the optimal masking ratio is unexpectedly high. Similarly, MAE [16] also reports high masking ratios, ideally ranging in 40%-80% for good fine-tuning performance on ImageNet-1K. For this dataset, MAE [16] produces similar results for ratio in range of 70%-90%, see Table 1. In contrast, the higher ratio achieves higher results in our case. We hypothesize that this behavior is connected to our method’s feature-level objective and adaptive masking as the mutual information is further maximized.

These ablated experiments justify the effectiveness of our method.

5.4 Effect of Adaptive Masking Strategy on Reconstruction.

In this section, we visualize and discuss the effect of the adaptive masking strategy on the reconstruction ability of our DAMA.

We show in Fig. 4 the reconstruction results of MAE [16] (b) and our five settings (c-g) similar to Table 4 on ViT Tiny and same masking ratio 80%. Except for (a), in (b-g), the upper row is the same random masking input $X^m_1$ (left) and its reconstruction result $G(Z^m_{X^m_1})$ (right), the lower row is the applied mask strategy input $X^m_2$ (left) and its reconstruction result $G(Z^m_{X^m_2})$ (right). From left to right, original image is center cropped as input green box (a), MAE
Fig. 3. Pre-training curves of different model settings on adaptive masking condition: Pixel-level reconstruction loss in network1 (a) and network2 (b), feature-level regression loss (c), and total loss (d). While pre-training without overlapping red curve would produce highest loss, share-weights and student1-student2 settings (green and purple curves, respectively) with overlapping perform better. Recall that overlapping means some unmasked patches in $X_{m1}$ will also be unmasked in $X_{m2}$.

[16] result (b), our model-ema random sampling result (c); our adaptive sampling (d-g): student-teacher without mask overlapping (d), student-teacher with mask overlapping (e), share-weights with mask overlapping (f), and student1-student2 with mask overlapping (g).

Without adaptive masking, (c) can not reconstruct properly. This could be explained as similarly to Model setting 5.3 ablation, the teacher network is constrained by its student network and could not produce reasonable inference with another random mask. In contrast, (d) is also in the student-teacher setting but applied with the adaptive mask is able to reconstruct the cell appropriately.

Note that, while MAE [16] can not reconstruct the region in red box which is part of another cell in a single iteration, our method can do that with the adaptive sampling. This suggests MAE [16] could leave out these fine-grain details even with several epochs in a random high masking ratio setting, i.e., 80%. Conversely, our DAMA combines pixel- and feature-level optimization with adaptive masking to identify those details in every iteration with a high masking ratio. This supports the advantage of our method.
Fig. 4. Reconstruction effect of mask sampling strategy between MAE [16] (b) and our (c-g). Except for (a), in (b-g), the upper row is same the random masking input $X^m_1$ (left) and its reconstruction result $G(Z_{X^m_1})$ (right), the lower row is the applied mask strategy input $X^m_2$ (left) and its reconstruction result $G(Z_{X^m_2})$ (right). From left to right, original image is center cropped as input green box (a), MAE [16] result (b), our student-teacher random sampling result (c); our adaptive sampling (d-g): student-teacher without mask overlapping (d), student-teacher with mask overlapping (e), share-weights with mask overlapping (f), and student1-student2 with mask overlapping (g). Note that, while MAE [16] can not reconstruct the region in red box in a single iteration, our method is able to do that with the adaptive sampling.

6 Conclusion

In the context of biomedical, where context is also a vital information resource, our method has shown potential results compared to the state-of-the-art on brain cells data by optimizing pixel-level reconstruction and feature-level regression. We report more accurate performance as increasing masking ratio as high as 90%. The proposed adaptive sampling positively impacts learning good representations and boosting performance. This could be the first adaptive masking method for masked image modeling to the best of our knowledge.
References

1. Agakov, D.B.F.: The im algorithm: A variational approach to information maximization. Advances in neural information processing systems 16(320), 201 (2004)
2. Baevski, A., Hsu, W.N., Xu, Q., Babu, A., Gu, J., Auli, M.: data2vec: A general framework for self-supervised learning in speech, vision and language. arXiv preprint arXiv:2202.03555 (2022)
3. Baevski, A., Zhou, Y., Mohamed, A., Auli, M.: wav2vec 2.0: A framework for self-supervised learning of speech representations. Advances in Neural Information Processing Systems 33, 12449–12460 (2020)
4. Bao, H., Dong, L., Wei, F.: Beit: Bert pre-training of image transformers. arXiv preprint arXiv:2106.08254 (2021)
5. Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J.D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al.: Language models are few-shot learners. Advances in neural information processing systems 33, 1877–1901 (2020)
6. 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)
7. Chen, T., Kornblith, S., Norouzi, M., Hinton, G.: A simple framework for contrastive learning of visual representations. In: International conference on machine learning. pp. 1597–1607. PMLR (2020)
8. Chen, X., Xie, S., He, K.: An empirical study of training self-supervised vision transformers. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 9640–9649 (2021)
9. Dai, Z., Yang, Z., Yang, Y., Carbonell, J., Le, Q.V., Salakhutdinov, R.: Transformer-xl: Attentive language models beyond a fixed-length context. arXiv preprint arXiv:1901.02860 (2019)
10. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition. pp. 248–255. Ieee (2009)
11. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)
12. Dmitrienko, A., Masiero, M.M., Zamboni, N.: Self-supervised learning for analysis of temporal and morphological drug effects in cancer cell imaging data (2021)
13. 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)
14. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. Advances in neural information processing systems 27 (2014)
15. 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.: Bootstrap your own latent-a new approach to self-supervised learning. Advances in Neural Information Processing Systems 33, 21271–21284 (2020)
16. He, K., Chen, X., Xie, S., Li, Y., Dollár, P., Girshick, R.: Masked autoencoders are scalable vision learners. arXiv:2111.06377 (2021)
17. He, K., Fan, H., Wu, Y., Xie, S., Girshick, R.: Momentum contrast for unsupervised visual representation learning. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 9729–9738 (2020)
18. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
19. Lin, Y., Gou, Y., Liu, Z., Li, B., Lv, J., Peng, X.: Completer: Incomplete multi-view clustering via contrastive prediction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11174–11183 (2021)
20. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3431–3440 (2015)
21. Maric, D., Jahanipour, J., Li, X.R., Singh, A., Mobiny, A., Van Nguyen, H., Sedlock, A., Grana, K., Roysam, B.: Whole-brain tissue mapping toolkit using large-scale highly multiplexed immunofluorescence imaging and deep neural networks. Nature communications 12(1), 1–12 (2021)
22. Van den Oord, A., Li, Y., Vinyals, O.: Representation learning with contrastive predictive coding. arXiv e-prints pp. arXiv–1807 (2018)
23. Radford, A., Narasimhan, K., Salimans, T., Sutskever, I.: Improving language understanding by generative pre-training (2018)
24. Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I., et al.: Language models are unsupervised multitask learners. OpenAI blog 1(8), 9 (2019)
25. Schneider, S., Baevski, A., Collobert, R., Auli, M.: wav2vec: Unsupervised pre-training for speech recognition. arXiv preprint arXiv:1904.05862 (2019)
26. Shen, H., Li, Y., Feng, M., Shen, X., Wu, D., Zhang, C., Yang, Y., Yang, M., Hu, J., Liu, J., et al.: Miscell: An efficient self-supervised learning approach for dissecting single-cell transcriptome. Iscience 24(11), 103200 (2021)
27. Tishby, N., Pereira, F.C., Bialek, W.: The information bottleneck method. arXiv preprint physics/0004057 (2000)
28. Tishby, N., Zaslavsky, N.: Deep learning and the information bottleneck principle. In: 2015 ieee information theory workshop (itw). pp. 1–5. IEEE (2015)
29. Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., Jégou, H.: Training data-efficient image transformers & distillation through attention. In: International Conference on Machine Learning. pp. 10347–10357. PMLR (2021)
30. Tsai, Y.H.H., Wu, Y., Salakhutdinov, R., Morency, L.P.: Self-supervised learning from a multi-view perspective. International Conference on Learning Representations (2020)
31. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems 30 (2017)
32. Vicar, T., Chmelik, J., Jakubicek, R., Chmelikova, L., Gumulec, J., Balvan, J., Provaznik, L., Kolar, R.: Self-supervised pretraining for transferable quantitative phase image cell segmentation. Biomedical optics express 12(10), 6514–6528 (2021)
33. Wu, M., Zhuang, C., Mosse, M., Yamins, D., Goodman, N.: On mutual information in contrastive learning for visual representations. arXiv preprint arXiv:2005.13149 (2020)
34. Xie, Z., Zhang, Z., Cao, Y., Lin, Y., Bao, J., Yao, Z., Dai, Q., Hu, H.: Simmim: A simple framework for masked image modeling. arXiv preprint arXiv:2111.09886 (2021)
35. Zbontar, J., Jing, L., Misra, I., LeCun, Y., Deny, S.: Barlow twins: Self-supervised learning via redundancy reduction. In: International Conference on Machine Learning. pp. 12310–12320. PMLR (2021)
36. Zhou, J., Wei, C., Wang, H., Shen, W., Xie, C., Yuille, A., Kong, T.: ibot: Image bert pre-training with online tokenizer. International Conference on Learning Representations (ICLR) (2022)
37. Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In: Proceedings of the IEEE international conference on computer vision. pp. 2223–2232 (2017)
A Appendix

Algorithm 1 Pytorch-like Adaptive Masking Pseudocode

```python
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
```

A.1 Implementation Details

Table A1. Pre-training (left) and fine-tune (right) setting of our DAMA.

| Config   | Value        | Config   | Value        |
|----------|--------------|----------|--------------|
| image size | 128×128      | image size | 128×128      |
| patch size  | 8×8          | patch size  | 8×8          |
| batch size  | 512          | batch size  | 512          |
| epochs      | 500          | epochs      | 150          |
| optimizer   | Adam         | optimizer   | Adam         |
| base learning rate | 1.5e-04   | Base learning rate | 1e-02   |
| min learning rate | 0        | min learning rate | 1e-5     |
| weight decay | 0.05        | weight decay | 0.05        |
| learning rate schedule | cosine decay | learning rate schedule | cosine decay |
| warmup epochs | 40          | warmup epochs | 5           |
| augmentation | RandomResizedCrop | augmentation | RandomResizedCrop |
| K-blocks/β  | 6/2          | droppath/reprob/mixup/cutmix | 0.1/0.25/0.8/1.0 |
Table A2. Pre-training (left) and fine-tune (right) setting of MAE[16].

| Config               | Value       | Config               | Value       |
|----------------------|-------------|----------------------|-------------|
| image size           | 128×128     | image size           | 128×128     |
| patch size           | 8×8         | patch size           | 8×8         |
| batch size           | 512         | batch size           | 512         |
| epochs               | 500         | epochs               | 150         |
| optimizer            | Adam        | optimizer            | Adam        |
| base learning rate   | 1.5e-04     | Base learning rate   | 1e-02       |
| min learning rate    | 0           | min learning rate    | 1e-5        |
| weight decay         | 0.05        | weight decay         | 0.05        |
| learning rate schedule | cosine decay | learning rate schedule | cosine decay |
| warmup epochs        | 40          | warmup epochs        | 5           |
| augmentation         | RandomResizedCrop | augmentation     | RandomResizedCrop |
| K-blocks/β           | 6/2         | droppath/reprob/mixup/cutmix | 0.1/0.25/0.8/1.0 |

Table A3. Pre-training (left) and fine-tune (right) setting of Data2Vec[2].

| Config               | Value       | Config               | Value       |
|----------------------|-------------|----------------------|-------------|
| image size           | 128×128     | image size           | 128×128     |
| patch size           | 8×8         | patch size           | 8×8         |
| batch size           | 512         | batch size           | 512         |
| epochs               | 800/1600    | epochs               | 150         |
| optimizer            | Adam        | optimizer            | Adam        |
| base learning rate   | 1.5e-04     | Base learning rate   | 1e-02       |
| min learning rate    | 0           | min learning rate    | 1e-5        |
| weight decay         | 0.05        | weight decay         | 0.05        |
| learning rate schedule | cosine decay | learning rate schedule | cosine decay |
| warmup epochs        | 40          | warmup epochs        | 5           |
| augmentation         | RandomResizedCrop | augmentation     | RandomResizedCrop |
| K-blocks/β           | 6/2         | droppath/reprob/mixup/cutmix | 0.1/0.25/0.8/1.0 |

Table A4. Pre-training (left) and fine-tune (right) setting of MoCo-v3[8].

| Config               | Value       | Config               | Value       |
|----------------------|-------------|----------------------|-------------|
| image size           | 128×128     | image size           | 128×128     |
| patch size           | 8×8         | patch size           | 8×8         |
| batch size           | 512         | batch size           | 512         |
| epochs               | 500         | epochs               | 150         |
| optimizer            | Adam        | optimizer            | Adam        |
| learning rate        | 1.5e-04     | Base learning rate   | 1e-02       |
| min learning rate    | 0           | min learning rate    | 1e-5        |
| weight decay         | 0.1         | weight decay         | 0.05        |
| learning rate schedule | cosine decay | learning rate schedule | cosine decay |
| warmup epochs        | 50          | warmup epochs        | 5           |
| augmentation         | RandomResizedCrop | augmentation     | RandomResizedCrop |
| stop-grad-conv1/moco-m-cos/moco-t | True/True/0.2 | droppath/reprob/mixup/cutmix | 0.1/0.25/0.8/1.0 |
**Table A5.** Setting of supervised scratch.

| Config              | Value          |
|---------------------|----------------|
| image size          | 128×128        |
| patch size          | 8×8            |
| 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 A6.** Pre-training (left) and fine-tuning (right) setting of DAMA on ImageNet-1k.

| Config                  | Value          | Config                  | Value          |
|-------------------------|----------------|-------------------------|----------------|
| image size              | 224×224        | image size              | 224×224        |
| patch size              | 16×16          | patch size              | 16×16          |
| batch size              | 4096           | batch size              | 1024           |
| epochs                  | 500            | epochs                  | 150            |
| optimizer               | Adam           | optimizer               | Adam           |
| base learning rate      | 1.5e-04        | Base learning rate      | 1e-02          |
| min learning rate       | 0              | min learning rate       | 1e-5           |
| weight decay            | 0.05           | weight decay            | 0.05           |
| learning rate schedule  | cosine decay   | learning rate schedule  | cosine decay   |
| warmup epochs           | 40             | warmup epochs           | 5              |
| augmentation            | RandomResizedCrop | augmentation          | RandomResizedCrop |
| K-blocks/β              | 6/2            | droppath/reprob/mixup/cutmix | 0.1/0.25/0.8/1.0 |
| mask ratio              | 0.8            |                          |                |
Fig. A.1. Example of five cell type images. From top to bottom: Iba1, NeuN, Olig2, RECA1, and S100.